
Z MIKRO-KOMPU-TEREM NA TY
2
Bajtek
MIESIĘCZNY DODATEK DO SZTANDARU MŁODYCH
NR 2/86
CENA 100 ZŁ
CZY MASZYNA MOŻE MYŚLEĆ!
str. 3

KRZEMOWA FALA

Spośród różnych przemawiających do wyobraźni objaśnień roli i znaczenia komputerów osobistych najbardziej podoba mi się metafora z rowerem. Jest otóż intuicyjnie i praktycznie oczywistym, że człowiek jadący rowerem — jeśli chodzi o efektywność wykorzystania energii mięśni — o wiele przewyższa wszystkie znane nam zwierzęta, gdy tymczasem człowiek bez roweru znajduje się wśród outsaiderów. Komputer osobisty to właśnie „rower dla umysłu" — indywidualny przyrząd wzmacniający naturalne możliwości ludzkiego mózgu.

Myślę, że powyższe porównanie najlepiej wyjaśnia dlaczego tak duży jest obecnie napór na wszystkie instytucje, mające cokolwiek wspólnego z upowszechnianiem komputerów osobistych. Piszę „instytucje", gdyż w grę wchodzi przecież drogi sprzęt, na który ktoś musi dać pieniądze, i za który ktoś musi odpowiadać.

Załóżmy, że nie mamy cioci za granicą, ani rodziców pracujących w handlu zagranicznym czy spółkach polonijnych, a chcemy — choćby dlatego, że przeczytaliśmy właśnie BAJTKA — nawiązać kontakt z komputerem osobistym. Co robić? Dokąd iść? Gdzie w ogóle szukać informacji na ten temat? Trudne pytania, ale odpowiedź na nie — wszystko na to wskazuje — będzie z dnia na dzień coraz łatwiejsza.

Rozwija się sieć klubów komputerowych Turnieju Młodych Mistrzów Techniki. Wojciech Wyszomirski — szef TMMT — marzy o tym, aby w każdym mieście, nawet małym, młody człowiek mógł iść czy też zadzwonić do tamtejszego Zarządu ZSMP i uzyskać informację:

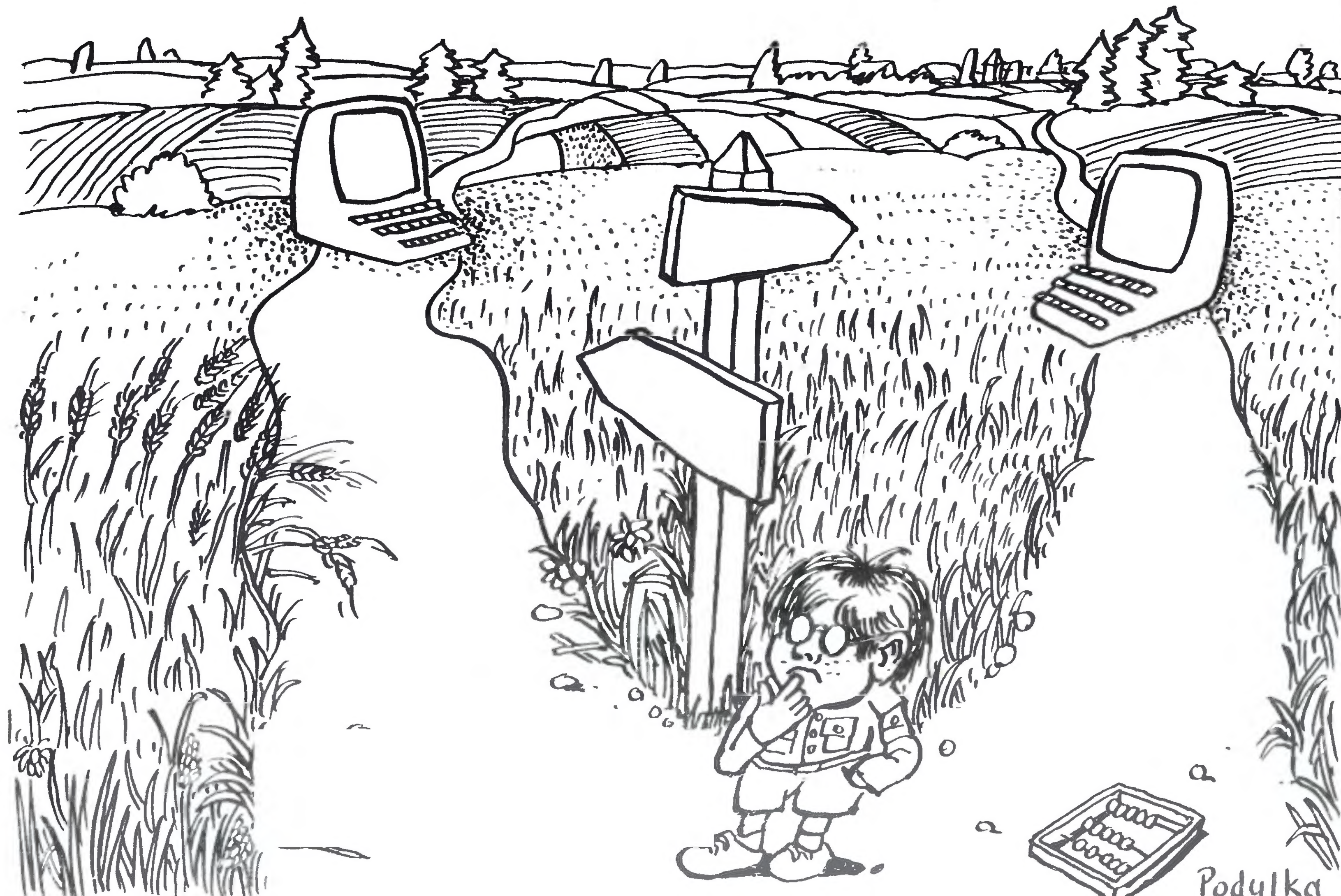
— Dobrze trafiłeś, właśnie u nas, codziennie można uzyskać dostęp do komputera osobistego!

Piękna to wizja i należy życzyć Wojtkowi wytrwałości w jej urzeczywistnianiu.

Rozwijają się kluby harcerskie organizowane w ramach systemu „InforMik". Druh Krzysztof Piotrowicz, zastępca naczelnika ZHP odpowiedzialny m.in. za sprawy twórczości naukowo-technicznej młodzieży, fanatyk nowoczesności, widzi w tych klubach jedną z najatrakcyjniejszych form kształtowania postaw twórczych, proinnowacyjnych i zapowiada szeroki rozwój harcerskiej informatyki. Pogratulować trzeba przy okazji kolegom z ZHP rozmachu — nie zdradzę tu tajemnicy, jeśli poinformuję, że Centralna Składnica Harcerska na serio przymierza się do sprowadzenia dużej partii komputerów osobistych i rozpoczęcia ich sprzedaży za złotówki, poniżej cen giełdowych. Brawo dyrektorze Szanter!

Rozwijaniem klubów komputerowych zajmują się studenci, spółdzielczość mieszkaniowa, zakłady pracy... Krzemowa fala płynie przez Polskę. Za jedno ze swych głównych zadań uważa BAJTEK ukierunkowanie tej fali, aby jej energia nie rozpraszała się w przebijaniu niepotrzebnie ustawionych falochronów, tylko zdołała pozostawić po sobie trwały edukacyjnie efekt.

Waldemar Siwiński



WYBIERZ SAM

„BAJTEK" — MIESIĘCZNY DODATEK DO „SZTANDARU MŁODYCH".

ADRES: 00-687 Warszawa, ul. Wspólna 61. Telefon 21-12-05.

Przewodniczący Rady Redakcyjnej: Jerzy Domański — redaktor naczelny „Sztandaru Młodych".

ZESPÓŁ REDAKCYJNY: Waldemar Siwiński (z-ca redaktora naczelnego „SM" — kierownik zespołu), Oskar Bramski, Sławomir Polak, Roman Poznański, Wanda Roszkowska (opr. graficzne), Roman Wojciechowski.

WYDAWCA: RSW „Prasa-Książka-Ruch" Młodzieżowa Agencja Wydawnicza, al. Stanów Zjednoczonych 53, 04-028 Warszawa. Telefony: Centrala 13-20-40 do 49, Redakcja Reklamy 13-20-40 do 49 w. 403, 414.

Cena 100 zł.

Skład techniką CRT-200, przygotowalnia offsetowa i druk: PRASOWE ZAKŁADY GRAFICZNE RSW „PRASA-KSIĄŻKA-RUCH" w Ciechanowie, ul. Sienkiewicza 51.

Zam. nr 349/86, nakład 200.000 egz. P-101

SM Bajtek



FOT. KRZYSZTOF WOJCIEWSKI

CZY MASZYNA MOŻE MYŚLEĆ

rozmowa z Andrzejem Gogolewskim, lat 33, inżynierem elektronikiem, absolwentem Politechniki Warszawskiej, twórcą układu sterowania najnowszego polskiego robota RIMP-900, wiceprezesem Młodzieżowej Akademii Umiejętności.

BAJTEK: — Czy maszyna może myśleć?

— Andrzej Gogolewski: — Jestem zwolennikiem poglądu, że inteligencja nie jest cechą wyłącznie ludzką. Choć dzisiaj jest to dla nas jeszcze nieosiągalne myślę, że uda się poznać i zrozumieć procesy myślowe. Gdy zaś je poznamy i zrozumiemy będziemy mogli przekazać je naszym tworom, czyli maszynom. Oczywiście nie muszą być to maszyny w dzisiejszym sensie tego słowa, mogą np. być zbudowane z substancji organicznych.

Człowiek jest na tyle ciekawy świa..., że będzie dążył do tego, aby zostać stwórcą. A co z tego wyniknie, trudno powiedzieć.

— Nie potrafimy jeszcze zdefi-

niować w ogóle pojęcia „inteligencja", a jednak dążymy przecież już dziś do stworzenia jej sztucznego odpowiednika. Na jakim etapie jesteśmy? Czy tradycyjna elektronika nie stanie wkrótce przed barierą nie do sforsowania?**

— Nie chciałbym prorokować co zwycięży — komputery organiczne, czy nieorganiczne. Obecnie próbuje się na świecie skonstruować komputer V generacji, będzie on miał dwie podstawowe cechy: równoległe przetwarzanie informacji i możliwość porozumiewania się przy pomocy języka naturalnego.

Fala eksperymentów z elastycznymi liniami produkcyjnymi i fabrykami bez ludzi doprowadziła do tego, że niemal wszystkie prace manipulacyjne zostały już przez automaty opanowane. Mogą one przeprowadzać nawet dość precyzyjny montaż. Natomiast do tej pory nie udało się nam zautomatyzować fazy prac projektowych, koncepcyjnych, choć i tu posługujemy się komputerami jako narzędziami. Sam proces twórczy nie został przyspieszony. Np. pisanie programów na komputer trwa mniej więcej tak długo, jak przed laty. Dlatego też oprogramowanie pozostaje w tyle za sprzętem. Gra idzie zatem o wysoką stawkę. Ten, komu uda się zautomatyzować procesy myślowe zyska od razu ogromną przewagę nad konkurentami, ponieważ będzie miał sprzężenie dodatnie w sprzęcie.

— Tak się dzieje w przypadku konstruowania komputerów.

— Istotnie. Chociażby mikroprocesory 32-bitowe nie są projektowane przez człowieka, lecz przez komputer przy pomocy ludzi. Kto ma zatem lepszy sprzęt i oprogramowanie dziś, może zapewnić sobie pierwszeństwo również na przyszłość.

Maszyny zaczynają także zastępować człowieka w procesie podejmowania decyzji. Jest to czasem użyteczne, czasem, jak w przypadku zbrojeń, groźne.

— Wróćmy jednak do komputerów V generacji. Czym będą się one różnić od dotychczas produkowanych pod względem architektury wewnętrznej?

— Jak wspomniałem komputer V generacji pracować będzie na zasadzie równoległego przetwarzania informacji. Trudno nam sobie nawet wyobrazić, jak będzie to wyglądało. Dotychczas, nawet w przypadku sprzętu wieloprocesorowego jest zawsze jeden nadrzędny procesor, który dzieli zadania. W przypadku komputera V generacji będziemy mieli sieć procesorów i potok danych z działaniami, które należy wykonać, płynący swobodnie przez tę sieć.

— Samo oprogramowanie będzie chyba zupełnie inne.

— Oczywiście. O ile wspomniane sieci procesorów już powstają, to oprogramowanie ich jest zabójcze.

Człowiek — w pewnym sensie — działa również na zasadzie równoległej. Prowadząc rozmowę jesteśmy w dalszym ciągu podatni na inne bodźce.

Równolegle do prac nad tworzeniem sieci mikroprocesorowych przeprowadza się doświadczenia z komputerami optycznymi. Tradycyjny komputer składa się de facto z tranzystorów. Układy małej czy też wielkiej skali integracji oznaczają jedynie upakowanie tych tranzystorów w jednej małej strukturze. Natomiast tranzystor optyczny, który został już skonstruowany działa jak bramka logiczna.

— Na razie jednak prace nad komputerami optycznymi należą do bocznego nurtu?

— Prace przyszłościowe bez względu na to na jak długo są obliczone powinny przynosić także efekty, nazwijmy to, etapowe. Nawet Japończycy, którzy postawili sobie za cel zbudowanie komputera V generacji też postawili sobie pewne cele cząstkowe. Sieć procesorów z pewnością powstanie szybciej niż komputer optyczny. Natomiast prowadzone są także prace nad komputerem biologicznym, czyli nad stworzeniem komórki, która wykorzystywałaby reakcje biochemiczne. Na ile są one zaawansowane? Nie wiadomo. Jeszcze dwa, trzy lata temu sporo się o tym pisało, dziś panuje cisza. To może oznaczać, że prace są bardzo bliskie celu. Dzieje się tak zresztą niemal z większością nowych rzeczy. Pisze się o nich, dopóki wydają się niemal niemożliwe do zrobienia, potem zapada cisza i wreszcie okazuje się, że to coś zostało już skonstruowane i działa.

— Wiadomo już o jaką konstrukcję z grubsza chodzi. A jak wygląda sprawa z komunikacją w języku naturalnym?

— Zakłada się, że wszystkie dotychczasowe konstrukcje zabrnęły trochę w ślepą uliczkę chociaż, moim zdaniem, znajomość języków programowania jest coraz bardziej powszechna.

— Ale jest ich coraz więcej i są one coraz mniej do siebie podobne.

— Jeśli chcemy nauczyć kogoś programowania powinniśmy abstrahować w ogóle od języka. Sprawa najważniejsza, to poznanie systemu myślenia algorytmicznego. Bardzo często wiemy, że trzeba coś zrobić, ale odpowiedź na pytanie od czego zacząć sprawia kłopoty. Konstruowanie algorytmów pomaga logicznie myśleć. Natomiast samo programowanie jest faktycznie czymś nienaturalnym. Nawet tzw. „myszki" czy też pióra świetlne są też nienaturalne. Dlatego dążymy do sytuacji, w której moglibyśmy opowiedzieć o problemie i otrzymać jego rozwiązanie. A to okazuje się bardzo trudnym zadaniem.

Rzecz nie jest nawet w samym rozumieniu słów, w pewnym stopniu to już opanowaliśmy. Powstały już translatory języków pracujące całkiem nieźle, tyle że w ograniczonym zakresie tematycznym. Jednak samo usłyszenie i zdekodowanie języka to jeszcze za mało. Usłyszane słowa trzeba jeszcze odnieść do rzeczywistości. Gdy rozmawiamy, rozumiemy się także dlatego, że dysponujemy podobną wiedzą i doświadczeniem. Mamy także podobne reguły wnioskowania. Dlatego, aby komputer nas zrozumiał powinniśmy przekazać mu prawie całą naszą wiedzę, stworzyć olbrzymią bazę danych, a także reguły wnioskowania. Jesteśmy dopiero na początku tej drogi. Obecnie potrafimy budować tzw. systemy ekspertów. Są to maszyny, które gromadzą wiedzę z danej dziedziny. Mogą one właśnie podejmować decyzje w przypadku awarii. Jest również wiele decyzji, które człowiek podejmuje niejako intuicyjnie, nie wiedząc dlaczego. I takie przypadki również podaje się komputerowi. Takie systemy powstały do dziś np. do stawiania diagnoz medycznych.

— Także do sterowania procesami produkcyjnymi?

— Wiemy, że na świecie stosuje się roboty przemysłowe i nie jest to żadną sensacją. Natomiast to, co pozwala nam mówić o kolejnej rewolucji przemysłowej to wprowadzenie tzw. elastycznych systemów produkcyjnych.

Wyobraźmy sobie kilka automatów wieloczynnościowych, przy nich roboty połączone siecią transporterów i wózków samojezdnych. Obraz ten uzupełnia magazyn również sterowany automatycznie. Wtedy możemy powiedzieć, że mamy do czynienia z elastycznym systemem produkcyjnym. Może się on w zasadzie obejść bez ludzi. Takie systemy pracują już w kilku zakładach. Dochodzi nawet do tego, że zautomatyzowane wózki akumulatorowe po wyczerpaniu baterii same zjeżdżają do warsztatu, gdzie roboty wymieniają im zużyte akumulatory. Jeżeli natomiast chcemy zmienić np. profil produkcji zakładu, to nie musimy nic przezbrajać. Przejście na nową produkcję wymaga jedynie przeprogramowania poszczególnych urządzeń i połączeń między nimi. Od tego już jeden krok do fabryki bez ludzi, tworu zamkniętego, żyjącego własnym życiem. Człowiek przekazuje jedynie informację o tym, co chciałby dostać na wyjściu i jakie podzespoły i materiały są do dyspozycji, a sam proces produkcyjny jest już optymalizowany przez system, który można określić jako duży system ekspertów.

— Jakie są zatem bariery powodujące, że dotychczas nie wprowadzało się na większą skalę takich zakładów?

— Dwie — ekonomiczna i społeczna. Osiągamy kolejny skok wydajności pracy i powstaje pytanie, co zrobić z tymi ludźmi, którzy przestają być zakładom potrzebni. Ten problem widać już na Zachodzie i powszechna jest świadomość, że musi on być rozwiązany.

Nauka o sztucznej inteligencji nie jest bynajmniej młoda. Jak dotąd nie udało się jednak odpowiedzieć na podstawowe pytanie: czy jesteśmy w ogóle w stanie poznać ludzki umysł? Być może człowiek nie może stanąć niejako obok siebie, po to, by przyjrzeć się obiektywnie swojej świadomości umysłowi, tak jak np. centymetrem nie sposób zmierzyć centymetr. Zachodzi tylko pytanie, czy jest to w ogóle potrzebne. Wszak koło również nie występuje w przyrodzie, a człowiekowi udało się przy jego zastosowaniu rozwiązać problem poruszania się. Może więc i sztuczna inteligencja nie będzie powielać przyrody.

Rozmawiał: Grzegorz Onichimowski

Czekając na powszechnie dostępny, konkurencyjny wobec zachodnich modeli polski mikrokomputer, mamy już dla niego elektroniczną sekretarkę — drukarkę D-100, produkowaną przez Zakłady Mechaniczne i Precyzyjne „Mera-Błonie".

Urządzenie, o rozmiarach maszyny do pisania, mieści się na biurku razem z monitorem komputera i klawiaturą. D-100 może znaleźć zastosowanie wszędzie tam, gdzie bezpośrednio lub poprzez urządzenie końcowe (terminal) istnieje łączność z komputerem. Zapisanie informacji na papierze ułatwia sprawdzanie i korygowanie obliczeń wykonywanych przez inżyniera, umożliwia sporządzenie historii choroby pacjenta itp.

Drukarka D-100 jest najmłodszą siostrą produkowaną w Błoniu drukarek komputerowych D-200 i D-180.

— *Są to różne unowocześnione odmiany tej samej serii tzw. drukarek mozaikowych* — wyjaśnia inż. **Tadeusz Dziewulski**, główny konstruktor w ZMP „Mera-Błonie". — *Każdy znak na papierowej taśmie składa się z pojedynczych punktów. Każdy z takich punktów, to uderzenie poprzez taśmę nasączoną tuszem, jednej z dziewięciu igieł umieszczonych w głowicy drukującej i poruszanych za pomocą elektromagnesu. Litery lub cyfry drukowane są szeregowo.*

Przed drukarką D-200, w latach siedemdziesiątych „Mera-Błonie" wykorzystywała francuską licencję na budowę drukarki komputerowej DZM-180. Obecnie najnowsza konstrukcja, czyli D-100 nie ma już nic wspólnego z licencyjnymi rozwiązaniami swojego pierwowzoru.

Błońska „detka" pozwala na uniknięcie ślęczenia z kartką papieru i długopisem przed migocącym ekranem monitora komputerowego. Umożliwia również wykonywanie rysunków, wykresów i tabel.

WIĘCEJ OZNACZA TANIEJ

Grupa krajowych właścicieli drukarek D-100 jest jeszcze niewielka. W zestawieniu z ceną domowego mikrokomputera, drukarka jest „względnie tania". Jeśli ktoś powiedział „a" stać go na kupno „b", co nie oznacza, że zrobi to łatwo, bez czekania, udowadniania po co i komu potrzebny jest ten luksus. Jednak dla szkoły, której np. podarowano kilka komputerów, kupno drukarki oznacza niebagatelny wydatek 150–180 tysięcy złotych.

Czas jednak biegnie naprzód, być może doczekamy się niższych cen. Mikrokomputery profe-

jonalne i domowe są produkowane w krajach RWPG już od kilku lat. W Zabrzu powstają „Meritum I" i „Meritum II". W Błoniu, oprócz drukarki montuje się w ciągu roku kilkaset mikrokomputerów „Mera-100 M". Wkrótce ma być ich więcej. Błońskie zakłady przystąpiły również do spółki krajowych przedsiębiorstw zgrupowanych pod szyldem „Mikrokomputery". Już w tym roku spółka zacznie montować pierwszą serię kilkuset polskich mikrokomputerów „Mazovia".

— *Będzie to sprzęt, o klasie porównywalnej do komputerów osobistych wytwarzanych przez amerykański koncern IBM* — zapewnia inż. **T. Dziewulski**. — *Rozwój produkcji różnych typów polskich mikrokomputerów stwarza krajowy popyt na naszą drukarkę. Planujemy więc, by w przyszłości produkować rocznie przynajmniej 50 tysięcy drukarek D-100. Taka ilość powinna nasycić rynek.*

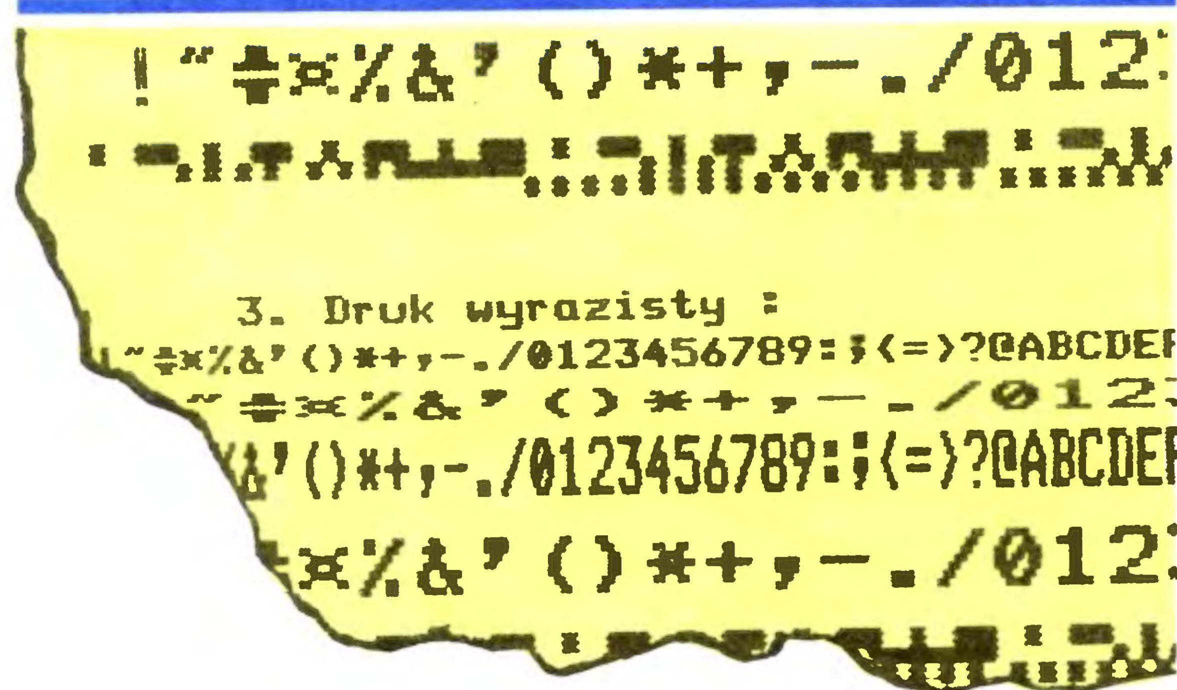
Dzisiaj wybór, czy D-100 trafia na rynek krajowy czy zagraniczny, rozstrzyga cena sprzedaży, która musi pokryć wciąż jeszcze wysokie koszty wytwarzania. Być może eksport nie zagłodzi krajowego rynku. Drukarka jest integralną częścią systemu komputerowego, tak jak kolumny głośnikowe są częścią zespołu odtwarzającego muzykę z płyty lub taśmy magnetofonowej. Oczywiście można się bawić bez głośników, ale wtedy impreza staje się już bardzo kameralna.

CO GŁOWA, TO GŁOWICA

W listopadzie ub.r. dwunastokilogramowa drukarka D-100 znalazła się na Światowej Wystawie Osiągnięć Młodych Wynalazców w Płowdiw. Zalety techniczne, lekkość i estetyczna obudowa spowodowały, że została obwołana jednym ze szlagierów bułgarskiej ekspozycji.

Drukarka D-100 podobnie jak jej starsze rodzeństwo, potrafi zapełnić papier pochyłym drukiem, zna różne kroje pisma i różne alfabety z arabskim włącznie. Wiele rozwiązań konstrukcyjnych w jej budowie chronią polskie patenty. Ich autorzy to ludzie młodzi. Właśnie dlatego błońska „detka" trafiła na wystawę w Bułgarii.

— *Po ukończeniu Wydziału Mechaniki Precyzyjnej Politechniki Warszawskiej, kilka lat temu znalazłem się w grupie ośmiu osób, które miały skonstruować drukarkę D-200* — wspomina inż. **Krzysztof Jeziorski**, konstruktor w ZMP „Mera-Błonie" — *Zacząłem od nauki, poznałem dokładnie zasadę działania drukarki komputerowej. Potem był upór, uzgadnianie szczegółów, wykonanie rysunków i wreszcie finał.*



Tempo prac i ich wynik zachęciły dyrekcję firmy do powierzenia młodemu inżynierowi kierowania pracami trzyosobowego zespołu, który miał skonstruować model głowicy drukarki D-100. Głowica drukująca licencyjnej drukarki DZM-180, ważącej 45 kilogramów była za duża dla planowanej zaledwie 12-kilogramowej D-100. Po roku została wykonana pierwsza seria drukarek z nowymi głowicami.

SŁOWA SPOD IGŁY



NOWE I UNOWOCZEŚNIANE

Jednocześnie z tworzeniem głowicy drukującej, inne zespoły inżynierów, techników i robotników opracowały pozostałe szczegóły konstrukcyjne D-100. Podobną technikę modernizowania produkowanego wyrobu w Błoniu stosowano już wcześniej, podczas unowocześniania drukarek wierszowych.

Drukarki mozaikowe D-100 w małych partiach docierają już do RFN, Francji i Anglii. Większość trafi do odbiorców w krajach socjalistycznych.

Błoński sukces ma wielu ojców. Mówią z dumą o ukryciu pod obudową drukarki mozaikowej nowinek technicznych będących odpowiedzią na rozwiązania narzucone światu przez firmy japońskie.

Inżynier **Krzysztof Jeziorski**:
— *W dalszym ciągu zmierzamy do tego, by konstrukcja urządzenia była maksymalnie prosta w wykonaniu. W drukarce D-100 znajduje się już np. uproszczony mechanizm przesuwający wstęgę papieru. W stosunku do poprzedniego modelu ograniczyliśmy zużycie energii elektrycznej.*

Marek Tkaczyk, technolog: — *Obecną, blaszaną obudowę drukarki D-100 chcemy zastąpić obudową z tworzyw plastycznych. Na razie wykonujemy próby. Być może wkrótce cena dostępnych w kraju specjalnych tworzyw będzie niższa, niż cena blachy. Sporządzenie plastykowej obudowy obniży więc koszt drukarki i skróci czas jej montowania.*

Konstruktorzy z „Mery" obiecują, że za trzy lub cztery lata drukarka D-100 będzie tańsza. Może będą już wówczas bardziej dostępne i tańsze polskie mikrokomputery? Dzisiaj „detka" zawiera bardzo drogie, w porównaniu do cen światowych, polskie podzespoły elektroniczne. Stąd też w 85 proc. produkcja „Mery-Błonie" wędruje poza granice Polski.

Wojciech Gładykowski


FOT. LEOPOLD DZIKOWSKI

Cezary Krajdener (z lewej) i Edmund Fliski: — *Być, to znać... komputer.*

SZKODA CZASU

— Ta mania komputerowa to błąd, który w przyszłości okrutnie się zemści! Ci ludzie są już straceni dla społeczeństwa...

— Co ty wygadujesz? To oni właśnie zadecydują, jaka będzie ludzkość. Kto tego nie rozumie — odpada. Stoimy dziś wobec konieczności przestawienia się na nowy typ myślenia.

Dyskusja staje się coraz bardziej gorąca. Tomasz Szcześniak z V „f" ze stoickim spokojem broni swojej tezy. Patrzy z góry na Dariusza Danielewskiego — też z piątej klasy. Przewaga wysokości zdaje się dawać mu także poczucie przewagi głoszonych poglądów.

— Przesada w zainteresowaniach powoduje, że ci ludzie nie widzą, co się wokół nich dzieje. Dla nich informatyka nie jest narzędziem, a celem. Nie można wyłącznie fascynować się mikrokomputerami. To już jest nałóg...

Gdy pada ten argument, małe pomieszczenie, będące „zapleczem" pracowni mikrokomputerowej, wypełnia fala śmiechu. Miłośnicy mikrokomputerów, których jest coraz więcej, mimo że wpadli tu na chwilę, okupują krzesła, stoją pod ścianami.

— Taka bezkrytyczna fascynacja jest tylko na początku — broni się Dariusz.

— Ładny początek, który trwa już drugi rok...

Potentaci

Zespół Szkół Elektronicznych im. PPR w Warszawie jest jedną z ośmiu stołecznych szkół, w których do programów nauczania wprowadzono techniki mikrokomputerowe. Do dyspozycji uczniów Meritum 2 ze stacjami dysków, Meritum 1 (48 kB — z rozszerzoną pamięcią), ZX Spectrum, Unipolbrity 2086, a także miniaturowe drukarki mozaikowe D-100. Jako monitory — rodzime „Neptuny". Wyposażenie dla koneserów może wydać się nierewelacyjne, ale... dla zdecydowanej większości szkół będące niedościgłym marzeniem. I to chyba jeszcze dość długo. Ze zrozumiałych względów nasuwa się pytanie: ile to wszystko kosztuje? Kto sfinansował zakup sprzętu?

— Po co o to pytać? Inni później denerwują się, że my, to tacy potentaci... — zastrzega się zastępca dyrektora d.s. techniczno-programowych mgr inż. Jolanta Szklarska.

Tylko część sprzętu jest własnością szkoły — osiem Meritum „wypożyczyło" Zrzeszenie MERA, zapewniając pełny serwis; dwa ZX Spectrum przekazało Ministerstwo Oświaty i Wychowania. Natomiast trzy Polbrity i ZX Spectrum Plus są własnością szkoły. W sumie, wartość sprzętu, który należy do placówki — ok. 2 mln zł. W zamian za korzystanie z wypożyczonych mikrokomputerów opracowywane są programy.

Dyrektor naczelny mgr inż. Stanisław Grefkowicz twierdzi, że są dziś dwa największe ograniczenia we wprowadzaniu edukacji mikrokomputerowej: finansowe i kadrowe. I tu — zaskoczenie, zwłaszcza po wymienionej liczbie mikrokomputerów, którymi dysponuje szkoła: dyrektor twierdzi, że znacznie większym ograniczeniem jest brak wykwalifikowanych nauczycieli niż... brak pieniędzy.

— Szkoły są w stanie zdobyć mikrokomputery własnym przemysłem, zaradnością. Będzie to jeszcze łatwiejsze, gdyż sprzęt będzie taniał. Poza tym zdobycie aparatury mogą ułatwić patronaty zakładów pracy.

O wiele gorzej jest ze zdobyciem nauczycieli. Czyżby wiązało to się z obiegową opinią, że to właśnie nauczyciele są najbardziej oporni na nowinki? Otóż nie! Nauczyciele chcą uczyć się informatyki i technik mikrokomputerowych — przynajmniej w tej szkole... Na kurs podstawowy, organizowany własnym sumptem, zapisali się — oprócz elektroników, matematyków czy fizyków, co wydaje się naturalne — m.in. pedagog, polonista, ekonomista, nauczyciele wychowania fizycznego. Szkoła istnieje piętnaście lat, ma profil elektroniczny — zawsze pracowali w niej nauczyciele znający informatykę. Łatwiej więc było zorganizować kurs, „przyuczać" innych.

— Jakimś dziwnym trafem dotarło do gazet, że szkoła robi takie kursy... telefony urywały się, drzwi się nie zamykały — wspomina dyrektorka nie bez odcienia dumy w głosie.

Ale tak dobrej sytuacji nie mają nauczyciele w małych miasteczkach — nie wspominając o wsiach. Zdaniem dyr. Grefkowicza, komputerową edukację nauczycieli trzeba zaczynać od dużych ośrodków miejskich, „bogatych w potencjał intelektualny".

Rozbudzanie

W pracowni elektroniki atmosfera pracy. Dwie dziewczyny biedzą się nad konstruowaniem wzmacniacza małej mocy. To ich praca dyplomowa.

— Szkoda będzie stąd odchodzić... Dlaczego? Dlatego, że to wszystko zaczęło się zbyt późno. Dopiero w tym roku mamy w programie techniki komputerowe, ale przecież chodzimy do piątej klasy, czeka matura, pogania praca dyplomowa... Za mało czasu, aby posiedzieć dłużej przed monitorem.

— Z komputerami jest tak — dodaje druga — najpierw na zasadzie nowinek, wszyscy tylko o tym rozmawiali, niektórzy chodzili do klubów. Wyjątkowo nieliczni mają własny sprzęt. Było to zainteresowanie powierzchowne. Część ludzi z czasem przestała się tym pasjonować. Ale gdy w szkole wystartował klub komputerowy, później pracownia — wszyscy znów rzucili się do maszynek.

Rzeczywiście, czas w pedagogice może być sprzymierzeńcem, ale także... wrogiem i to dość bezwzględnym. Zgodę na wprowadzenie przedmiotu Ministerstwo Oświaty i Wychowania wydało już kilka lat temu. Informatyka jest w szkole przedmiotem obowiązkowym. Ale były to wykłady prowadzone „na sucho".

— Wyszliśmy z założenia, że nie wolno dłużej tracić czasu. — mówi dyr. Szklarska — opracowaliśmy wewnętrzny program, który — jak się później okazało — był bardzo zbieżny z propozycjami Ministerstwa. Udało się zgromadzić sprzęt i — nie bez obaw, oraz ryzyka — zaczęliśmy. To był najlepszy moment, bo rozbudzenie zainteresowań wśród młodzieży osiągnęło temperaturę wrzenia.

— Od tego roku, każdy absolwent z dyplomem technika elektronika — bez względu na profil klasy — ma za sobą przeszkolenie informatyczne. Kształcenie trwa rok, obecnie jest prowadzone w III, IV i V klasie oraz na III i IV semestrze w studium policealnym.

W szkole działa także klub mikrokomputerowy. Gdy go otwierano, było tak dużo chętnych, że okazała się potrzebna selekcja. Aby była sprawiedliwa, jedynym kryterium był... test na inteligencję.

Być to znać... komputer

Dzwonek obwieszcza przerwę. Pomieszczenie pustoszeje — w pracowni przez chwilę zostaje dwóch chłopaków, wyraźnie nie dostrzegających końca zajęć. Rozmawiają przyciszonymi głosami.

— Dasz do przegrania swój, a ja ci dam... „Gwiezdne wojny". Nie chcesz? Może więc „Boa", „Otello", „Video", „Pyton", „Kombajn", „Wieże Hanoi", „Zbijanie muru", „Robak Franciszek", „Maszyna do pisania"...?

— ...

— To jeszcze czystą kasetę dorzucę...

— W porządku.

A tymczasem w pomieszczeniu będącym zapleczem pracowni mikrokomputerowej trwa dyskusja na temat celu i sensu zainteresowań informatyką, technikami mikrokomputerowymi. Zahacza o filozofię, informatykę i.... złotówki oraz waluty tzw. obce.

Cezary Krajdener: Jestem pasjonatem Meritum... trochę z konieczności. Rozpracowałem tajniki oprogramowania, ale coraz częściej dochodzę do wniosku, że Meritum przy Spectrum to jak „syrenka" przy „fordzie"...

Edmund Fliski: Informatyka jest takim działem, że po dwóch, trzech latach nauki bez styczności z komputerem — wszystko traci sens... Dlatego nie będę zdawał na informatykę. Wybiorę taki kierunek, na którym będę nadal bawić się komputerami.

Olgierd Dudko: Moje marzenie jest oczywiste: komputer. Jakie są szanse? Teoretycznie jakieś są... będę chodził do szkoły dwadzieścia lat, pracował w czasie wakacji i ferii... Na pewno odłożę sto tysięcy na maszynkę.

Wojciech Mróz: W szkole podstawowej interesowałem się fotografią, chciałem zrobić zegar ciemniowy. Tak zaczęło się zainteresowanie elektroniką, do tego doszła informatyka.

●

— Mówiłam, że nie wolno tracić czasu? Oni są przerażająco zdolni. Tak, proszę właśnie tak napisać: przerażająco zdolni — mówi dyrektorka na pożegnanie.

Roman Wojciechowski

CO TO JEST EFEKT MAGNETRONOWY?

Prezentowany program napisany na mikrokomputer ZX SPECTRUM stanowi ilustrację do ciekawego zjawiska fizycznego, zwanego efektem magnetronowym. Zrealizowany w formie zabawy ułatwia zrozumienie praw fizyki.

Jeżeli cząsteczka o masie „m" i ładunku elektrycznym „e" wpadnie z prędkością „v" prostopadle w jednorodne pole magnetyczne „B" to tor jej ruchu zacznie się zakrzywiać. Gdy pole będzie wystarczająco duże, cząsteczka może zostać uwięziona w nim, gdyż będzie się stale poruszała po tym samym okręgu o promieniu „R" z częstotliwością kątową „w". Płaszczyzna okręgu jest prostopadła do kierunku pola magnetycznego. Jest to właśnie efekt magnetronowy.

Na tak uwięzioną cząsteczkę działają dwie siły: siła oddziaływania pola magnetycznego na naładowaną, poruszającą się cząsteczkę

$$F_1 = e\,v\,B$$

i siła odśrodkowa

$$F_2 = \frac{m\,v^2}{R}$$

które się równoważą

$$F_1 = F_2$$

stąd

$$R = \frac{m\,v}{e\,B}.$$

Posługując się prezentowanym programem można zaobserwować w jaki sposób początkowe parametry „v" i „B" wpływają na kołowy ruch cząstki (opisywany parametrami „R" i „w"). Najpierw wybieramy prędkość wstrzelenia cząsteczki w pole (w granicach 0, 1-2 w jednostkach umownych). Początkowa wartość indukcji pola magnetycznego wynosi 1 gaus (1 Gs). Obserwując ruch cząsteczki w polu możemy wartość pola stopniowo (co 0,2 Gs) zwiększać aż do 3 Gs. Aktualne parametry ruchu cząsteczki („R" i „w") są podawane w dolnym, prawym rogu ekranu, aktualna wartość indukcji magnetycznej pola w górnym, lewym rogu, a prędkości — w górnym prawym (rys. 1). Cząsteczka na ekranie porusza się po torze eliptycznym, ponieważ elipsa jest obrazem koła w rzucie ukośnym. Uzyskuje się w ten sposób przestrzenność modelu. Klawisze biorące udział w sterowaniu animacją oraz ich funkcje są wypisane w dolnym lewym rogu monitora.

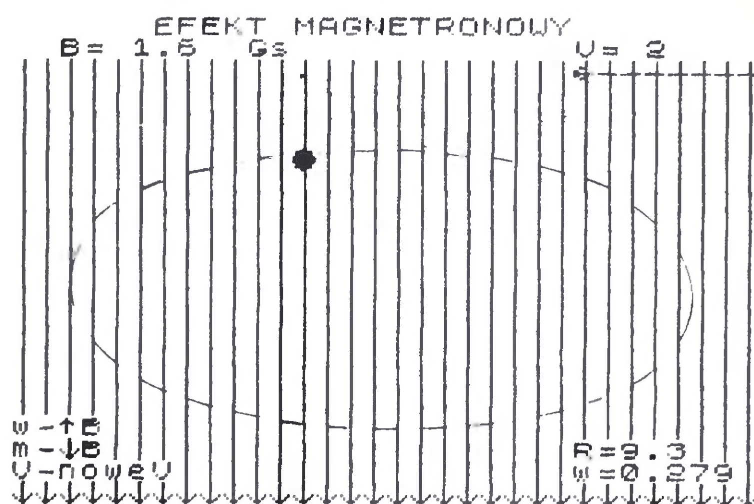
Na początku zdefiniowano (w wierszach 12-20) trzy znaki graficzne: elektron — strzałką skierowaną w dół i pionową kreską. Są one uzyskiwane po **wciśnięciu w trybie graficznym liter** (odpowiednio **P S i K) (UWAGA: znaki można wywoływać po wczytaniu wierszy 12-20** i uruchomieniu programu).

Podprogram zaczynający się od etykiety 900 jest blokiem informacyjnym krótko omawiającym zjawisko efektu magnetronowego z wyprowadzeniem wzoru na wartość e/m tzn. na stosunek ładunku elektrycznego cząsteczki do jej masy. Jest to wielkość ważna, bo charakterystyczna dla danego rodzaju cząsteczek (inna jest dla elektronów, inna dla protonów czy cząstek). Dlatego wyznaczenie wartości e/m pozwala na identyfikację cząsteczki. Na tej zasadzie działa jeden z najprecyzyjniejszych analizatorów fizykochemicznych, a mianowicie spektrograf masowy.

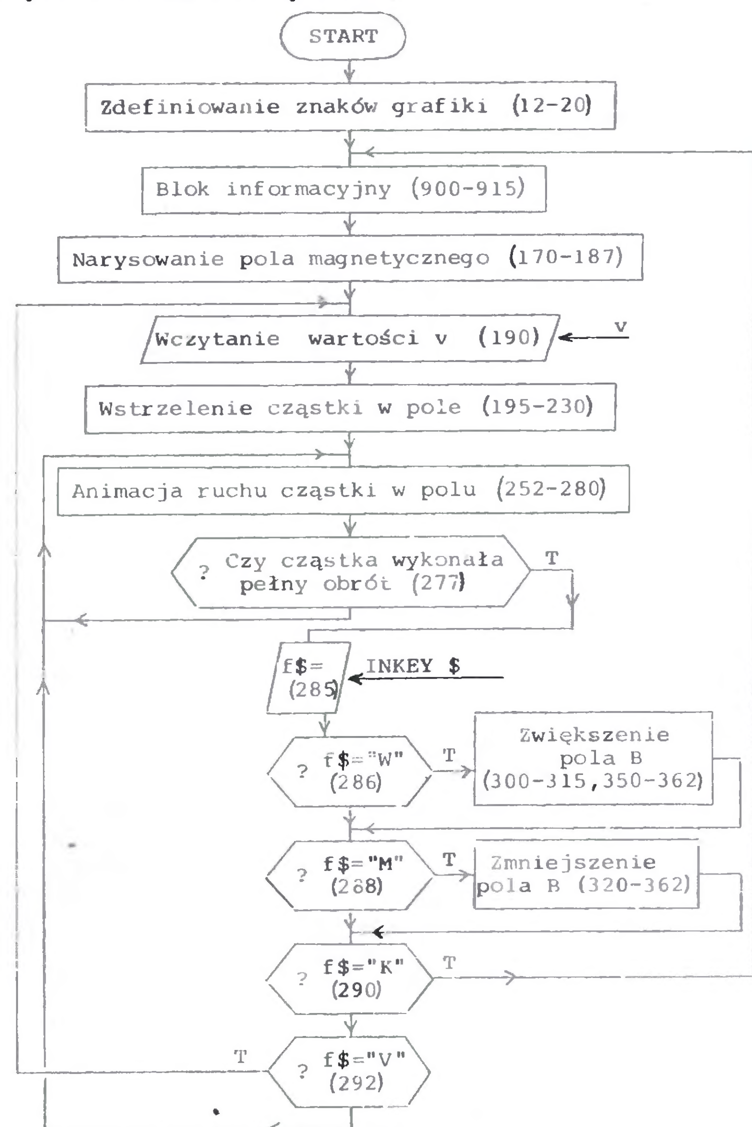
Gdy zapoznaliśmy się z teoretycznymi podstawami efektu magnetronowego, możemy nacisnąć ENTER i zacząć zabawę. Po wczytaniu prędkości „v" cząsteczka wpada w pole magnetyczne i zaczyna w nim krążyć. Możemy wpływać na jej ruch zwiększając wartość pola (przez przyciśnięcie klawisza W) lub zmniejszając ją (przez przyciśnięcie klawisza M). Przyciśnięcie klawisza V przerywa ruch cząstki i umożliwia nadanie jej innej początkowej prędkości. Istnieje możliwość powrotu do bloku informacyjnego przez przyciśnięcie klawisza K. Należy jednak pamiętać, że powyższe przełączenia są realizowane przez program tylko wówczas, gdy cząsteczka jest w górnym położeniu ekranu (patrz schemat blokowy programu).

I teraz już wiemy wszystko! Ale czy rzeczywiście? A jak zależy prędkość obrotu cząsteczki „w" od wartości pola magnetycznego „B" i od prędkości początkowej „v"? Gdy uruchomisz program na pewno to odgadniesz. Przyjemnej zabawy.

Tomasz Kostrzyński



Rys. 1 Efekt magnetronowy na monitorze. Linią przerywaną zaznaczono tor cząsteczki.



Rys. 2 Schemat blokowy programu.

```
   5 REM EFEKT MAGNETRONOWY
  10 REM DEFINICJE ZNAKOW GRAFIC
ZNYCH
  12 LET u=BIN 00010010: LET g=B
IN 00011000: LET h=BIN 01111110:
 LET i=BIN 11111111: LET j=BIN 0
0001000
  13 FOR f=1 TO 3
  14 READ p$
  15 FOR n=0 TO 7
  16 READ b: POKE USR p$+n,b: NE
XT n
  17 NEXT f
  18 DATA "p",g,h,h,i,i,h,h,g
  19 DATA "k",j,j,j,j,j,j,j,j
  20 DATA "s",j,j,j,j,BIN 010010
01,BIN 00101010,BIN 00011100,j
  21 GO SUB 900
  22 REM ANIMACJA

 170>PAPER 0: INK 7: BORDER 0: C
LS
 175 PRINT TAB 6;"EFEKT MAGNETRO
NOWY"
 176 PRINT ""
 180 FOR n=1 TO 19
 182 PRINT "||||||||||||||||||||
|||||||||||||": NEXT n
 184 PRINT "↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓
↓↓↓↓↓↓↓↓↓↓↓"
 187 PRINT AT 2,31;"●";AT 1,2;"B
= 1     Gs"
 190 INPUT "Wybierz V (0.1<V<2)"
;v
 195 LET wo=PI/36
 197 IF v<=0.1 OR v>2 THEN LET v
=1
 200 LET w1=v*wo
 202 PRINT AT 1,24;"V= ";v
 204 LET ao=15: LET b=9
 206 PRINT AT 19,24;"R=";ao;AT 2

 207 PRINT AT 18,0;"w-↑B": PRINT
 "m-↓B": PRINT "V-noweV"
 208 LET w=w1: LET a=ao: LET to=
0
 209 LET y=2: LET x=15: LET Bm=1
 210 PLOT 255,155
 212 FOR n=1 TO 16
 215 PRINT AT 2,32-n;"|"
 220 IF n<=8 THEN DRAW -8,0
 222 IF n=8 THEN DRAW 5,3: DRAW
-6,-3: DRAW 6,-3
 225 PRINT AT 2,31-n;"●"
 227 BEEP .1,n
 230 NEXT n
 235 LET t=0
 252 LET xa=INT (-1*a*SIN (w*t)+
15.5)
 255 LET ya=INT (-1*b*COS (w*t)+
11.5)
 260 PRINT AT y,x;"|"
 265 PRINT AT ya,xa;"●"
 270 LET y=ya: LET x=xa

 272 BEEP .001,12
 275 LET t=t+1
 277 IF w*t>2*PI THEN LET t=0: G
O TO 285
 280 GO TO 252
 285 LET f$=INKEY$
 286 IF f$="w" THEN GO SUB 300
 288 IF f$="m" THEN GO SUB 320
 290 IF f$="k" THEN GO TO 21
 292 IF f$="v" THEN PRINT AT ya,
xa;"|";AT 1,5;"1  ";AT 1,27;"
";AT 20,26;"|||||";AT 19,26;"||
|||";AT 2,31;"●": GO TO 190
 295 GO TO 252
 300 LET Bm=Bm+.2
 310 IF Bm>3 THEN LET Bm=3
 316 GO TO 350
 320 LET Bm=Bm-.2
 325 IF Bm<1 THEN LET Bm=1
 350 LET a=ao/Bm: LET w=Bm*w1
 355 LET b=.6*a: LET to=1
 360 PRINT AT 1,5;Bm;"  ";;AT 20
,26;INT (1000*w)/1000;"|";AT 19,
26;INT (10*a)/10;"||"
 362 RETURN
 400 REM BLOK INFORMACYJNY

 900>PAPER 7: INK 0: BORDER 7: C
LS
 901 PRINT "  Sila dzialajaca ze
strony polamagnetycznego na cza
stke o ladunku  e  poruszajacej
sie prosto- padle do wektora ind
ukcji magne-tycznej  B  z predko
scia  v jestokreslona przez wyra
zenie :"
 902 CIRCLE 35,155,6: CIRCLE 76,
139,5: CIRCLE 211,139,5
 903 PRINT AT 6,12; INVERSE 1;"
F = e v B "; INVERSE 0
 904 PRINT "Jest ona rowna sile
odsrodkowej:"
 905 INVERSE 1: PRINT AT 8,20;"2
"
 906 PRINT AT 9,12;" F = m v / r
": INVERSE 0
 907 PRINT ""
 909 PRINT "   stad:"
 910 PRINT "              e   v"

 911 PRINT "              m  r B
"
 912 PLOT 110,71: DRAW 10,0: PLO
T 135,71: DRAW 24,0: PLOT 126,72
: DRAW 5,0: PLOT 125,70: DRAW 5,
0: PLOT 102,59: DRAW 0,26: DRAW
68,0: DRAW 0,-26: DRAW -68,0
 913 PRINT AT 15,1;"Aby wyznaczy
c "; INVERSE 1;"e/m"; INVERSE 0;
" nalezy znac nastepujace wielk
osci fizyczne: "; INVERSE 1'AT 1
7,8;"v";AT 17,16;"B";AT 17,24;"r
"; INVERSE 0
 914 INPUT "Nacisnij ENTER";f$
 915 RETURN
 916 REM
```

PROGRAMOWAĆ MOŻE KAŻDY

Tym razem rozmawiać będziemy o wielokrotnym zadawaniu pytań złożonych, o rekurencyjnym definiowaniu relacji oraz o pewnych własnościach list i o tym, co z tych własności wynika.

Załóżmy, że bardzo często pytamy o dziadka ze strony ojca. Naturalnie od razu narzuca się nam następująca struktura pytania:

which (x y:x father-of z and z father-of y).

Pary (x y) będące odpowiedziami na tak postawione pytanie to

Henry-Snr Mary
Henry-Snr Elizabeth2
Henry-Snr Edward

Relację występującą między tymi osobami moglibyśmy nazwać **„paternal-grandfather-of"**. Jej definicja byłaby następująca:

x paternal-grandfather-of y wtedy i tylko wtedy, gdy dla pewnych z zachodzą: x father-of y i jednocześnie y father-of z.

W Prologu zaś napiszemy tak:

x paternal-grandfather-of y if x father-of z and z father-of y

Teraz Prolog, żeby rozwiązać zadanie typu:

which (x y: x paternal-grandfather-of y)

musi znaleźć argumenty spełniające logiczną koniunkcję zdań:

x father-of z and z father-of y

Równie prosto możemy zdefiniować relację: **parent-of** (rodzic). Jest chyba oczywiste, że:

x parent-of y if x father-of y or x mother-of y

Dotychczas w budowaniu definicji używaliśmy „zdań warunkowych". Ogólnie mają one postać:

A if C_1 and C_2 and C_3 and ... and C_n

gdzie C_1, C_2, C_3 ... C_n są warunkami. Zdanie A nazywamy wówczas konsekwencją zdań po prawej stronie implikacji. Przypominam, że zdanie A jest prawdziwe (ma logiczną wartość TRUE) wtedy, gdy prawdziwe są zdania: C_1, C_2, C_3 ... C_n.

Nowo zdefiniowane relacje mogą oczywiście zawierać relacje uprzednio zdefiniowane. Np.:

x grandparent-of y if x parent-of z and z parent-of y

x jest dziadkiem (lub babcią) y, jeśli x jest ojcem (lub matką) z i jednocześnie z jest ojcem (lub matką) y.

Definicje w Prologu możemy układać w bardzo wyszukany sposób. Na przykład używając rekursji.

Jeżeli mamy bardzo dużą rodzinę i chcemy znaleźć na przykład wszystkich przodków Mary, nie wiedząc jak wielu ich może ona mieć, zmuszeni jesteśmy do zastosowania procedury rekurencyjnej. Definicja:

„Twoi przodkowie to twoi rodzice i przodkowie twoich rodziców" jest rekurencyjna, gdyż w swojej definicji odwołuje się do pojęcia, którego jest definicją.

Konstrukcja zrozumiała dla Prologu może wyglądać następująco:

x ancestor-of y if x parent-of y
x ancestor-of y if z parent-of y and x ancestor-of z

Gdy zapytamy teraz system o przodków Edwarda:

which (x: x ancestor-of Edward)

odszukani zostaną rodzice Edwarda, następnie rodzice jego rodziców i tak dalej...

Definicji rekurencyjnych możemy używać również do tworzenia funkcji matematycznych. Zdefiniujemy silnię liczby N. Jak wiemy, silnia liczby N jest to iloczyn wszystkich liczb naturalnych mniejszych lub równych N.

N! = 1 * 2 * 3 * ... * (N-1) * N

definicja rekurencyjna składa się z dwóch części:

1 silnia 1
x silnia y if 1 LESS and SUM (x1 1 x) and x1 silnia y1 and TIMES (x y1 y)

Ażeby znaleźć taki y, że x! = y dla danego x Prolog: sprawdza czy x nie jest równy jeden /wtedy y = 1/, jeśli nie, to: odejmuje 1 od x, by otrzymać x1, następnie szuka takiego y1, że x1 = silnia y1 i mnoży y1 przez x, aby otrzymać y. Cała procedura jest powtarzana aż do momentu, w którym x1 osiągnie wartość 1.

Zdefiniujemy jeszcze jedną bardzo przydatną procedurę:

between (pomiędzy), która będzie wyszukiwała liczby z przedziału (y x).

y pomiędzy (y z) if y LESS z
x pomiędzy (y z) if SUM (y 1 y1) and y1 LESS z and x pomiędzy (y1 z).

*

Jeżeli dana relacja występuje między pewnymi grupami obiektów, to używamy wtedy w Prologu pojęcia listy. Wprowadźmy relację parents-of /rodzice/:

(ojciec matka) parents-of (wszystkie ich dzieci)

Mamy teraz zależność między dwiema listami danych: ojcem i matką oraz ich dziećmi. Zapiszmy relację:

(Henry Sally) parents-of (Margaret Bob)
(Henry Mary) parents-of (Elizabeth Bill Paul) (Bill Jane) parents-of (Jim)
(Paul Jill) parents-of (John Janet)

Teraz odpowiedzią na pytanie:

which (x: (Henry y) parents-of x)

będą następujące listy danych:

(Margaret Bob)
(Elizabeth Bill Paul)

Możemy również zapytać o wszystkie takie rodziny, które posiadają troje dzieci i dwoje rodziców.

all /children x1 x2 x3 father x mather y: /x y/ parents-of /x1 x2 x3/ /

i odpowiedzią będzie następująca lista:

(children Elizabeth Bill Paul father Henry mother Mary)

Jeżeli chcemy bardziej „zagłębić" strukturę zmiennych, możemy stosować listę jako element listy. Przykładem może być lista list i elementów:

((a b c) () ((d) e))

() — oznacza listę zeroelementową.

Chcąc zgromadzić więcej danych o jednym elemencie możemy ułożyć listę następującą:

(X x1 x2 x3 ... xn) np:

((Bill 53) (Jane 47) (parents-of) (Jim 17))

jest przykładem relacji pomiędzy listami rodziców i dzieci.

Z kolei każda z osób i jej wiek tworzą również listę.

Reprezentacja listowa nie zawsze jest wygodna. Zdefiniujmy relację **y mother-of-child x** (y matką dziecka x)

y mother-of-child x1 if x parents-of x1 x2
y mother-of-child x2 if x y parents-of x1 x2

Odnosi się ona wyłącznie do rodziny z dwojgiem dzieci. Problem ten możemy rozwiązać stosując nieco odmienną definicję:

y mother-of-child z if (x y) parents-of z and z belongs-to Z

gdzie Z jest listą dzieci. Przypominam, że małymi literami oznaczamy elementy a dużymi listy.

Kłopot w tym, że nie potrafimy jeszcze zdefiniować relacji belongs-to. Aby to uczynić, musimy wpierw poznać strukturę listy.

Pierwszy element listy nazywany jest głową listy /head/ zaś cała reszta /drugi, trzeci... n-ty/ — ogonem listy /tail/. Głową listy /A/ jest A, a ogonem ().

Dwa prawa określają przynależność danego elementu do liczby:

— głowa listy jest elementem listy
— dowolny element należy do listy, jeśli należy do jej ogona.

Oczywiście ogon listy jest równie listą, a więc też ma swoją głowę.

Zapis: (x/y)

oznacza listę, której x jest głową zaś Y — ogonem. Pionowa kreska oznacza, że Y jest listą elementów następujących po x1... xn /następnikiem/.

Jeszcze kilka bardziej złożonych przykładów:

((x y) (Z)) oznacza listę, która zawiera dwuelementową podlistę np.: dla ((a b) c d) x = a, y = b. Z = (cd)

((x Y)Z) określa podlistę jednoelementową, której następnikiem jest pewna lista.

Jeśli nie znamy liczby elementów listy. którą chcemy zapisać, możemy jej ostatnie składniki przedstawić również w postaci listy. W ogólnym przypadku będzie to wyglądało tak:

(x1 x2 ... xn Y)

Teraz możemy już zdefiniować rekurencyjnie relację belongs-to

x belongs-to (x Z) /1/
x belongs-to (y Z) if x belongs-to Z /2/

Spróbujmy przeanalizować jak Prolog będzie odpowiadał na pytanie

all (x: x belongs-to (A B C D))

Na początku otrzymano odpowiedź, została ona odnaleziona przy pomocy definicji /1/ przez podstawienie

x belongs-to (A B C D E)

Następnie posługując się definicją /2/ znajduje następną zmienną, spełniającą zależność

x belongs-to (B C D E)

i historia się powtarza aż do rozwiązania pozostaje ostatnia zależność

x belongs-to ()

Ponieważ w liście pustej () nie ma żadnych elementów, więc odpowiedzią jest:

No (more) answers. Nie ma (więcej) odpowiedzi.

Oto kolejna, bardzo przydatna relacja: ma — długość, wiążąca ze sobą dwa obiekty: listę i jej długość. Definiujemy ją rekurencyjnie:

() ma-długość 0
(x X) ma-długość z if X ma-długość y and SUM (y 1 z)

Sens tej definicji jest następujący: lista pusta ma długość zero, zaś każda lista niepusta ma długość o jeden większą niż jej ogon.

O długość listy (A B C D) pytamy:

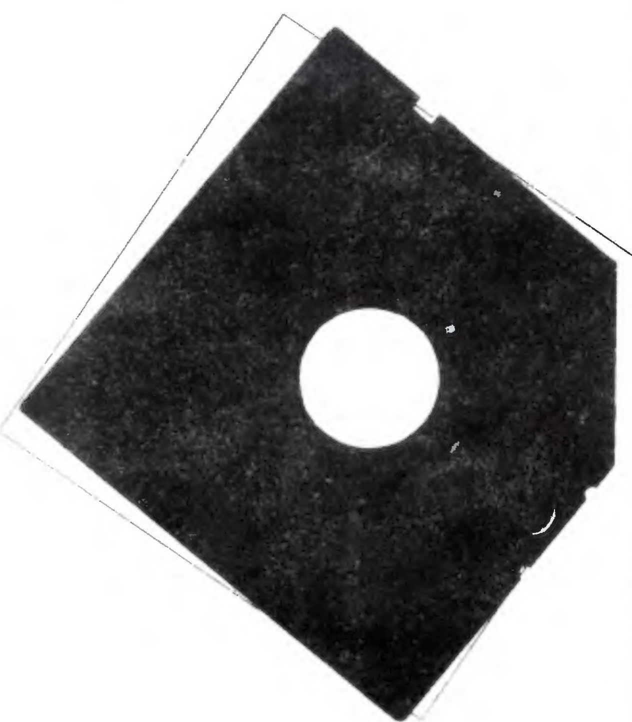
which (x: (A B C D) ma-długość x)
4
No (more) answers

By zaś sprawdzić czy długością tej listy

PROGRAMOWANIE W JĘZYKU LOGIKI cz. 2



PROGRAMOWANIE W JĘZYKU LOGIKI cz. 2



jest 4 piszemy:

((A B C D) ma-długość 4)

Zdefiniujmy jeszcze relację odwrotną

O długością ()

y długością (x X) if y INT and O LESS y and

SUM (z l y) and z długością X

Używamy tej relacji chcąc znaleźć listę o zadanej długości:

which (x : 4 długością X)

(A B C D)

No (more) answers

W Prologu występuje wiele warunków złożonych. Jednym z nich jest **isall** /wszystko/. Jej użycie powoduje znalezienie wszystkich odpowiedzi do danego pytania, oraz przedstawienie ich w postaci listy obiektów. Tak więc pytanie:

which (X : X isall (y : Henry father-of y))

ma odpowiedź następującą:

(Edward Elizabeth2 Mary)

No (more) answers

Możemy w ten sposób określać pewne definicje dotyczące wszystkich obiektów o danej cesze. Najprostszą taką relacją jest father-of-children /ojciec dzieci/

x father-of-children Y if Y isall (z : x father-of z)

Powyższą definicję interpretujemy następująco: Obiekt x jest ojcem listy Y dzieci wtedy, gdy Y jest listą wszystkich takich obiektów z, że zachodzi następujący związek: x jest ojcem z.

Ogólnie syntaktyka isall jest następująca: L isall (A : C), czyli L jest listą takich A, że C zachodzi dla niektórych y1, y2, ... yk Zmienne y1, y2, ..., yk są lokalne dla warunku C. Bardzo wygodnie jest używać warunku isall w połączeniu z na przykład niedawno utworzoną relacją „należy-do"

Definicja:

X częścią-wspólną (Y Z) if X isall (x:x należy-do Y and x należy -do Z)

jest spełniona wtedy gdy X jest listą tych wszystkich obiektów, które są elementami listy Y oraz listy Z.

Prolog dysponuje zestawem operatorów arytmetycznych:

=, +, -, ∗,

Znak „ = " ma podobne znaczenie co relacja EQ. Różnica polega na tym, że gdy rozwiązywany jest warunek „ = " zmiennym zostaną nadane wartości zanim zostaną one porównane relacją EQ. W przypadku:

x = (y 67 + z)

zmiennej x zostanie nadana wartość wyrażenia w nawiasie. Operator równości może działać również na listach danych czyli w wyrażeniu:

(x y) = (2 ∗ z) (z) 45()

x przyjmie wartość (2 ∗ z), zaś y (z) 45

Równość

24 = (2 ∗ x ∗ x + 7 ∗ x)

może być użyta do sprawdzenia czy jakiś x spełnia równanie:

$2x^2 + 7x = 24$

Nie może zaś być użyta do znalezienia pierwiastków tego równania. Możemy jedynie sprawdzić czy liczby z danego przedziału je spełniają. Do tego celu używamy także uprzednio zdefiniowanej relacji „pomiędzy".

Pytamy więc:

which (x : x pomiędzy (1 24) and 24 = (2 ∗ x ∗ x + 7 ∗ x))

Syntaktyka jak się mogliśmy zorientować jest postaci:

(wyrażenie operator wyrażenie)

gdzie operatorem jest jeden ze znaków:

∗ dla mnożenia

% lub / dla dzielenia

\+ dla dodawania

\- lub ˜ dla odejmowania.

Następną formą wyrażeń są wywołania funkcji. Przypuśćmy, że relacje div i modsą zdefiniowane przez następujące reguły:

div(x y z(if TIMES)y z1 x(and INT)z1 z)

mod(x y z) if div(x y z1) and z = (x--y ∗ z1)

Jak się domyślamy relacja div reprezentuje dzielenie całkowite /dzielną jest x a podzielnikiem z/.

Mod służy do znalezienia reszty z dzielenia x przez y.

Tak utworzone relacje różnią się od innych tym, że ostatni argument jest funkcją dwu poprzednich. By zasygnalizować systemowi odmienny charakter div i mod piszemy:

function div

function mod

Wywołanie funkcji jest listą postaci: (R E1 E2 .. En-1), gdzie E1, E2, .. En-1 są wyrażeniami a R nazwą relacji. R musi być zdeklarowana jako funkcja komendą:

function R

Funkcji używamy w programie w następującej formie:

R(V1 V2 .. Vn-1 x),

gdzie V1, V2, .. Vn-1 są wartościami odpowiednich wyrażeń E1, E2, .. En-1

W przypadku niezdeklarowania funkcji kompilator wyśle wiadomość:

R assumed not be a function co w wolnym przekładzie oznacza: „Przypuszczam, że R nie jest funkcją".

Możemy w każdej chwili przeczytać listę dotychczas zdeklarowanych funkcji komendami:

which (x : x func)

lub

list func

Funkcja może być argumentem dowolnego operatora arytmetycznego.

Istnieje jeszcze jeden operator „ # " używany przy połączeniach relacji z wyrażeniami zawierającymi operatory arytmetyczne i funkcje. Np.:

LESS # ((2 ∗ X) (5 + y))

jest relacją o argumentach, będących wartościami wyrażeń odpowiednio:

(2 ∗ X), (5 + y)

Syntaktyka jest następująca:

R # (E1, E2, ... En)

gdzie R jest nazwą relacji, a E1, E2, ... En wyrażeniami „ # " sygnalizuje, że E1, E2, ... En nie są normalnymi argumentami, gdyż niektóre z nich, lub wszystkie zawierają operatory arytmetyczne i funkcje.

Równość E1 = E2 możemy zapisać:

EQ # (E1 E2)

Jak widać zmienia dane wyrażenie na jego wartość.

Możemy więc używać tego operatora do podręcznych obliczeń. Np.:

&. # (3 ∗ 5 + 9)

24

Pytanie takie jest alternatywne do pytania:

which (x : x = (3 ∗ 5 + 9))

Generalną formą takich zapytań jest:

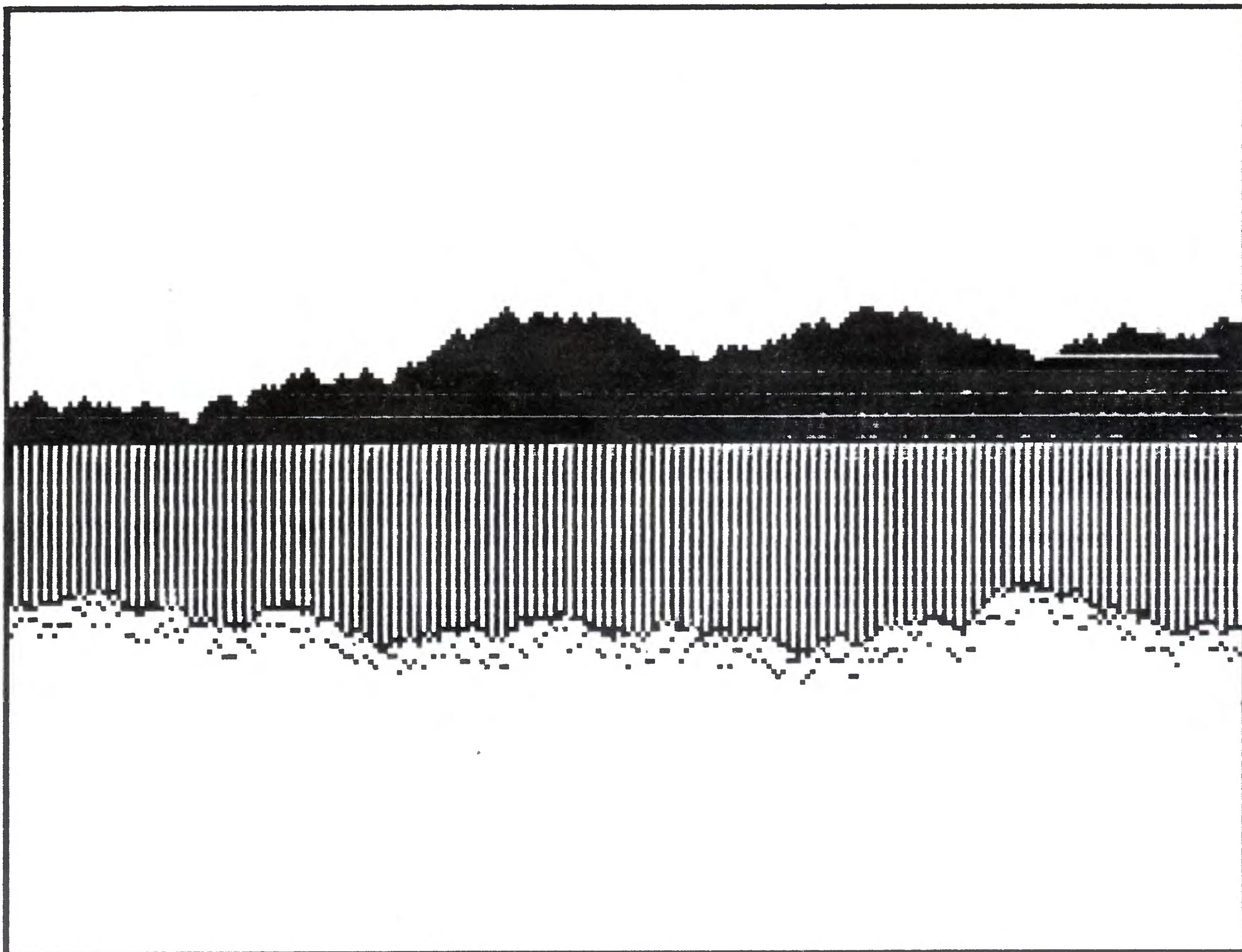
\# ‹wyrażenie›

Jeszcze jednym często stosowanym operatorem jest „ /∗ ", którym oznaczamy komentarz.

/∗ komentarz

Adam Krauze (lat 18)

```
1 REM RUCHOMY KRAJOBRAZ
5 BORDER 0: PAPER 7: INK 0: BRIGHT 1: CLS
10 LET X=255: LET Y1=79: LET Y2=120
15 GOSUB 1000: REM PODPROGRAM WPROWADZAJACY KOD MASZ.
20 LET C1=0: LET C2=0
50 LET R1=INT (RND*2)+1: REM TWORZENIE GOR
60 IF R1=1 THEN LET Y1=Y1+1
70 IF R1=2 THEN LET Y1=Y1-1
80 IF Y1<55 THEN LET Y1=55
90 IF Y1>111 THEN LET Y1=111
100 PLOT X,Y1: REM RYSOWANIE BIALYCH GOR
105 LET C2=C2+1:IF C2=2 THEN LET C2=0:DRAW 0,111-Y1: REM RZEKA
110 LET R2=INT (RND*6)+1
120 PLOT X,Y1-R2: REM STOKI BIALYCH GOR
130 LET D=USR 40000: REM PRZESUNIECIE SRODKOWEJ CZESCI EKRANU
133 LET C1=C1+1:IF C1=3 THEN LET C1=0: GOSUB 200
140 GOTO 50
200 LET R3=INT (RND*2)+1 : REM TWORZENIE GOR
210 IF R3=1 THEN LET Y2=Y2+1
220 IF R3=2 THEN LET Y2=Y2-1
230 IF Y2>175 THEN LET Y2=175
240 IF Y2<112 THEN LET Y2=112
250 PLOT X,Y2 : REM RYSOWANIE CZARNYCH GOR
255 DRAW 0,11[illegible] Y2
257 POKE 4000[illegible],71: REM ZMIANA KODU MASZYNOWEGO
260 LET D=USR 40000: REM PRZESUNIECIE GORNEJ CZESCI EKRANU
263 POKE 40002,79: REM PRZYWROCENIE PIERWOTNEGO KODU
270 RETURN
1000 FOR T=40000 TO 40016
1010 READ A
1020 POKE T,A: REM WPROWADZENIE KODU MASZ. POD ADRES 40000
1030 NEXT T
1040 RETURN
1045 REM KOD MASZ. DO PRZESUWANIA W LEWO SRODKOWEJ CZESCI EKRANU
1050 DATA 33,255,79,14,64,6,32,183,203,22,43,16,251,13,32,245,201
```



Program „Ruchomy krajobraz” wykorzystuje operację przesuwania ekranu w lewo o jeden punkt. Ekran ZX Spectrum składa się z trzech poziomych części po 64 linie. Przesuwając osobno każdą z nich można uzyskać ciekawe efekty.

W środkowej części ekranu, w ostatniej kolumnie (255) rysowany jest punkt. Następnie zostaje wywołany program w języku maszynowym powodujący przesunięcie całej środkowej części w lewo. W 255 kolumnie rysowany jest znowu punkt o pionowej współrzędnej większej lub mniejszej o jeden (jest to uzależnione od wartości obliczonej przez funkcję RND generującą liczby pseudolosowe) od poprzedniego punktu, który znalazł się teraz w kolumnie 254.

Powtarzając ten cykl uzyskujemy na ekranie przesuwające się pasmo „gór”. Wszystko to działo się w środkowej części ekranu i ona tylko była przesuwana. Jeśli tak samo zrobimy w górnej części ekranu i jeśli na każde trzy przesunięcia części środkowej przypadnie jedno przesunięcie części górnej, to uzyskamy perspektywiczny widok dwóch pasm górskich oglądanych z jadącego pojazdu (góry w tle przesuwają się wolniej niż góry w przednim planie). Aby uczynić krajobraz bardziej realistycznym, pomiędzy dwoma łańcuchami górskimi rysowana jest rzeka, a pasmo znajdujące się bliżej ma przypadkowo ułożone stoki.

A oto jak wygląda w asemblerze program maszynowy zapisany jako w linii 1050 pod instrukcją DATA

```
       ld hl,20479   ;adres końca przesuwanej części
       ld c,64       ;liczba linii w tej części
poz 1  ld b,32       ;liczba bajtów w linii
       or a
poz 2  rl (hl)       ;przesunięcie w lewo
       dec hl        ;następny bajt w linii
       djnz poz2     ;ta linia skończona ? nie — skok do poz 2
       dec c         ;następna linia
       jr nz, poz1   ;wszystkie linie przesunięte ? — nie — skok do poz 1
       ret           ;powrót do BASICa
```

W pierwszej linii do rejestru hl wstawiany jest adres końca środkowej części. Aby przesunąć część górną należy adres ten zmienić na 18431, co wykonywane jest w linii 257 programu w BASICu przez zmianę pierwszego bajtu adresu na 71.

Michał Szuniewicz
(lat 16)

RUCHOMY KRAJOBRAZ

ZEGAR COMMODORE

Komputery Commodore posiadają wewnętrzny realizowany programowo zegar czasu rzeczywistego. Zegar zostaje uruchomiony po włączeniu komputera i startuje od wartości zerowej. Przy włączeniu komputera wartość czasu ginie.

Rozkaz PRINT TI$ powoduje wydruk czasu. Wydruk ten występuje w formacie **ggmmss**, gdzie gg to godzina mm — minuty, ss — sekundy. Np. godzina **15** minut **32** sekund **45** wyglądać będzie następująco 153245.

Rozkazem **10 INPUT TI$** możemy ustawić aktualny czas w formie ggmmss.

Rozkaz **SYS 65499** zeruje zegar.

Rozkaz **POKE 788,50** zatrzymuje zegar.

Poniższy program umożliwia ciągły wydruk czasu.

```
10 INPUT TI$
20 PRINT CHR$ (147)
30 PRINT "[■]", TI$
40 GOTO 30
```

Znak [■] to naciśnięcie klawiszy SHIFT i CRSR.

Zegar synchronizowany jest częstotliwością z sieci.

Komputery Commodore posiadają też licznik TI zwiększający swoją zawartość co 1/60 sekundy.

Spróbujmy napisać program.

```
10 PRINT CHR$ (147)
20 PRINT "[■]",, TI
30 GOTO 20
```

Po uruchomieniu tego programu otrzymujemy wydruk zawartości licznika TI. Licznika TI nie możemy ustawiać rozkazem INPUT. Możemy go tylko zerować rozkazem SYS 65499.

Marek Bednarczyk
student PW

WSZYSTKO DLA WSZYSTKICH

Pewnie znalazłeś już przeglądając ten numer planszę i opis gry „Hobbit" i powoli nabierasz chęci, aby spróbować swoich sił w tym programie. Musisz wiedzieć, że „Hobbit" należy do najbardziej skomplikowanych programów jakie wymyślono dla mikrokomputera.



W grze Twój mikrokomputer pełni rolę pośrednika. Aby wydawać polecenia i uzyskać informacje dotyczące gry, musisz nauczyć się posługiwać językiem INGLISH.

Język INGLISH powstał w wyniku naturalnej ewolucji prowadzącej do zminimalizowania wiedzy niezbędnej do komunikowania się z komputerem. Jak dotychczas jego zastosowania w szerszej skali nie powiodły się ze względu na ograniczone możliwości swobodnej konwersacji (ograniczony słownik i brak możliwości formułowania abstrakcyjnych pojęć).

Język ten nadaje się jednak świetnie w grach mikrokomputerowych, co wykorzystali twórcy oprogramowania mikrokomputerowego tworząc nowy typ gier tzw. ADVENTURE (przygodowych).

Zasady INGLISH są b. proste. Jeżeli znasz podstawy języka angielskiego, znasz już INGLISH, choć jeszcze o tym nie wiesz.

Podstawą konstrukcji zdania w INGLISH jest czasownik.

Najkrótsze zdanie — to sam czasownik. Na przykład:

RUN biec
CLIMB wspinać się
WAIT czekać

Znaczenie słów może być zmienione przez użycie np. przysłówków

RUN QUICKLY
biegnij szybko

Jak i w normalnej gramatyce tak i tu szyk wyrazów w zdaniu można zmieniać.

WITH THE SWORD CAREFULLY ATTACK THE TROLL.
ATTACK THE TROLL CAREFULLY WITH THE SWORD.

pomimo różnego szyku znaczenie jest to samo:
zaatakuj ostrożnie trola mieczem.

Przymiotniki, które opisują przedmioty, muszą stać przed rzeczownikami.

OPEN THE GREEN DOOR — dobrze
OPEN THE DOOR GREEN — źle

Jest to zupełnie proste. Jeśli zdania rażą Cię, możesz być pewien, że i komputerowi nie spodobają się.

Przyimki takie jak: WITH (z), UNDER (pod), ON (na), OFF (poza) zazwyczaj stoją przed rzeczownikiem

ATTACK WITH THE SWORD
atakuj mieczem
PICK UP THE GOLD
weź złoto.

Choć jest kilka przyimków, które mogą występować po rzeczownikach, np.:

TURN THE LIGHT ON
zapal światło
PICK THE GOLD UP
weź złoto.

Tak też można.

ZASTOSOWANIE AND

AND używasz w języku INGLISH we wszystkich jego normalnych znaczeniach. Oznacza to między innymi, że możesz wprowadzić naraz więcej niż jedno zdanie

TAKE THE LAMP AND THE ROPE OUT OF THE BARREL
zabierz lampę i linę z beczki
DROP THE SHORT AND THE LONG SWORD
wyrzuć krótki i długi miecz
TAKE THE MONEY AND RUN
zabieraj pieniądze i uciekaj

ZNAKI PRZESTANKOWE

Różne zdania powinny być rozdzielone przy pomocy znaków przestankowych: stosuj AND, przecinki, średniki oraz kropki tak, jak w normalnej pisowni.

Jedynym ograniczeniem nałożonym przez komputer dotyczącym ilości wprowadzanych naraz zdań jest ograniczenie do 128 znaków.

Oczywiście, pamiętaj, że w każdej chwili gdy Ty coś robisz, inne stworzenia mogą też decydować co robić i możesz zostać zaskoczony niespodziewanymi wynikami, gdy wprowadzisz naraz zbyt dużo poleceń.

STOSOWANIE ALL, EVERYTHING I EXCEPT

Będzie dla Ciebie niewygodne opisywanie każdej rzeczy w pomieszczeniu, gdy chciałbyś wziąć wszystko.

Masz możliwość uogólnienia poprzez użycie ALL (wszystko), EVERYTHING (wszystko), i EXCEPT (oprócz).

Możesz określić czym chcesz się w danej chwili posługiwać w sposób ALL BARRELS (wszystkie beczki) lub możesz powiedzieć EVERYTHING EXCEPT GREEN BOTTLES (wszystko oprócz zielonych butelek).

Przykłady zdań zbudowanych prawidłowo:

EAT EVERYTHING
zjedz wszystko
OPEN ALL EXCEPT THE GREEN BOTTLE
otwórz wszystko oprócz zielonej butelki
BREAK ALL BOTTLES EXCEPT THE GREEN ONE
stłucz wszystkie butelki oprócz zielonej

OGRANICZENIA W INGLISH

Aby opisać przedmiot, możesz używać tylko nazwy danego przedmiotu i określającego przymiotnika (jeśli taki jest). Na przykład jeśli widzisz, że w butelce jest trochę wspaniałego piwa z pianką, możesz powiedzieć:

DRINK BEER
wypij piwo
lub DRINK DELICIOUS BEER
wypij wspaniałe piwo
lub DRINK FOAMING BEER
wypij piwo z pianką
lub DRINK DELICIOUS FOAMING BEER
wypij wspaniałe piwo z pianką
lub DRINK FOAMING DELICIOUS BEER

Nie musisz też określać miejsca, w którym akurat znajduje się przedmiot przez Ciebie opisywany np. zdanie:

DRINK BEER IN BOTTLE
wypij piwo z butelki

nie jest prawidłowe (aby wypić piwo nie musisz określać, że jest ono w butelce).

Nie możesz również stosować w zdaniu więcej niż jednego dopełnienia. Oznacza to, że nie możesz polecać w zdaniu zrobienia jednej rzeczy w więcej niż jeden sposób. Możesz powiedzieć:

PUT THE ROPE ON THE TABLE
połóż linę na stole

Możesz zadecydować, aby położyć ją na krześle

PUT THE ROPE ON THE CHAIR (na krześle)

Ale nie możesz powiedzieć:

PUT THE ROPE ON TABLE AND CHAIR
połóż ją na stole i krześle

Jest to ogólna zasada i oznacza ona też, że nie możesz powiedzieć w następujący sposób:

ATTACK THE WOLVES WITH EVERYTHING
zaatakuj wilki wszystkim

Wydaje nam się, że powyższe ograniczenia są bardzo małe i że nie przeszkodzą Ci w wyrażaniu się w sposób dla Ciebie najwygodniejszy.

OKREŚLANIE KIERUNKU

Aby określić kierunek ruchu możesz użyć klawiszy oznaczonych strzałkami (można to robić tylko na początku zdania), oraz używając pełnych nazw kierunków, skrótów lub całych zdań.

np.

E
EAST
GO EAST
GO E
RUN EAST
QUICKLY GO EAST

Wszystkie te formy są poprawne.

Możesz wprowadzić więcej niż jedno zdanie i użyć kierunków jako części dłuższego polecenia w następujący sposób:

TAKE EVERYTHING AND GO EAST
weź wszystko i idź na wschód

lub

TAKE ALL, E
weź wszystko, wschód

STOSOWANIE SKRÓTÓW

Z komputerem można porozumiewać się w sposób krańcowo skrótowy i zawsze będzie się dobrze zrozumianym.

Należy jednak uważać, bo w przypadku znacznych skrótów, może nastąpić nieporozumienie.

Jest także możliwe pisanie mniej niż pełnych zdań i jeżeli nie ma w nich błędów, komputer zrobi to co chcesz.

np.

Jesteś w komfortowym tunelu (norze) jak korytarz.
Na wschodzie są zielone drzwi.
Widzisz: drewnianą skrzynię.
Załóżmy, że chcesz otworzyć drzwi i mówisz:
OPEN THE GREEN DOOR
otwórz zielone drzwi.

Możesz powiedzieć krótko

OPEN

GLISH

otwórz
Lecz komputer nie wie, co otworzyć (można skrzynię i drzwi).
Pyta:
OPEN WHAT?
co otworzyć?
Masz możliwości wyboru odpowiedzi.
Możesz podać nazwę przedmiotu lub odpowiedzieć:
ALL
wszystko
Komputer przetwarza twoją instrukcję i w „okienku przygody" widzisz wyniki swojego polecenia:
YOU OPEN THE GREEN DOOR
otwierasz zielone drzwi
YOU OPEN THE WOODEN CHEST
otwierasz drewnianą skrzynię
Gdybyś znów wprowadził polecenie OPEN (otwórz), komputer odpowie Ci bardzo rozsądnie:
Nie widzę niczego, co mógłbym otworzyć.
I SEE NOTHING TO OPEN.
Zauważ, że gdy napiszesz pełne zdanie
Otwórz drzwi.
Odpowiedź będzie następująca:
Zielone drzwi są otwarte.
Jest to cena, którą płacisz za to, że wyrażasz się skrótowo.

KOMUNIKOWANIE SIĘ Z INNYMI POSTACIAMI

W języku INGLISH możemy również wydawać polecenia innym stworzeniom biorącym udział w grze.
Zasada jest wyjątkowo prosta:
Wprowadzasz:
SAY TO (powiedz) WHOMEVER (komuś) „sentence" (zdanie)
np.
SAY TO GANDALF „READ MAP"
Powiedz Gandalfowi „przestudiuj mapę"
Wiadomość, którą chcesz przekazać, musisz napisać w cudzysłowie i normalnie stosować zasady języka INGLISH.

SŁOWNIK

SPECJALNE POLECENIA

LOOK (spójrz — skrót L) Daje ci pełny opis miejsca, w którym jesteś, wszystkich wyjść, widocznych przedmiotów, poza tymi, które niesiesz.
INVENTORY (spisz — zinwentaryzuj — skrót I) — Daje ci opis wszystkiego, co niesiesz.
EXAMINE object: (zbadaj przedmiot): Umożliwia bliższe spojrzenie na przedmiot, z którym możesz się zetknąć.
WAIT: (czekaj): pozwala na przepływ czasu.
@ : (nie wymaga wciśnięcia klawisza ENTER).
Jest to wyjątkowo wygodny klawisz, który wprowadza polecenie: „Wykonaj ostatnie polecenie jeszcze raz" — DO LAST COMMAND AGAIN".
PRINT (drukuj): Polecenie to przekazuje obraz na drukarkę (jeśli połączona jest z Twoim komputerem). Jest to bardzo wygodne przy rozwiązywaniu problemów gry.
NOPRINT (nie drukuj): to polecenie uniemożliwia drukowanie.
SAVE (przechowaj): polecenie to umożliwia przechowanie informacji o Twojej przygodzie na taśmie do momentu wznowienia gry. Gdy gra zostanie zapisana w pamięci, komputer może zapytać Cię, czy chcesz ją obejrzeć jeszcze raz dla weryfikacji. Gdy zakończysz zapisywanie części gry, dalszą grę możesz prowadzić normalnie.
Możliwość przechowania w pamięci komputera jest bardzo przydatna, gdy gra jest zaawansowana (zawędrowałeś daleko przez Wilderland) a nie chcesz zaczynać od początku, gdzie mógłbyś zostać przypadkowo zabity.
LOAD: Polecenie przekazania z taśmy wcześniej wprowadzonych przygód
QUIT: Zaczyna grę ponownie.
SCORE (stan gry): Podaje Ci jaki procent gry masz za sobą.
PAUSE: zatrzymuje grę do momentu naciśnięcia dowolnego klawisza.
HELP: Pozwala Ci uzyskać wskazówki dotyczące dalszego prowadzenia gry.

RUCH

DOWN D	— w dół
EAST E	— wschód
NORTH N	— północ
NORTHEAST NE	— płn.-wsch.
NORTHWEST NW	— płn.-zach.
SOUTH S	— południe
SOUTHEAST SE	— płd.-wsch.
SOUTHWEST SW	— płn.-zach.
UP U	— w górę
WEST W	— zachód

DZIAŁANIE

BREAK	— łamać
CLIMB	— wspinać się
CLOSE	— zamykać
CROSS	— przechodzić
DIG	— kopać
DROP	— wypuszczać, rzucać, powalić
DRINK	— pić
EMPTY	— opróżniać
ENTER	— wchodzić
EAT	— jeść
FILL	— napełniać
FOLLOW	— iść za
GIVE	— dawać
GO	— iść
KILL	— zabić
LOCK	— zamykać na klucz
PICK UP	— podnosić
PUT	— położyć
OPEN	— otwierać
RUN	— biec
SAY	— mówić
SHOOT	— strzelać
SWIM	— pływać
TIE	— wiązać
THROW	— rzucać
TAKE	— brać
TURN	— przekręcać
UNLOCK	— otwierać kluczem
UNTIE	— rozwiązać
WEAR	— ubierać

PRZYIMKI

ACROSS	— przez
AT	— w, na
FROM	— z
IN	— w
INTO	— do
ON	— na
OUT	— poza, na zewnątrz
OFF	— poza
TO	— do
THROUGH	— przez
UP	— w górę
WITH	— z

PRZYSŁÓWKI

CAREFULLY	— ostrożnie
GENTLY	— delikatnie
QUICKLY	— szybko
SOFTLY	— łagodnie
VICIOUSLY	— źle, złośliwie, gwałtownie

Sławomir Polak





PROGRAMOWAĆ MOŻE KAŻDY

Jednym z założeń twórców języka Logo było zbliżenie sposobu komunikacji z komputerem do języka naturalnego. Udało się to w dużej mierze i jakkolwiek cecha ta najbardziej jest widoczna dla języka angielskiego (nie ma w nim koniugacji i deklinacji) to jest możliwe również znalezienie wspólnych cech Logo i języka polskiego.

Naszkicujmy kilka takich podobieństw.

Odnieśmy układ programista — komputer to znanej nam z młodych lat opowieści o Kopciuszku. Rolę komputera, czy inaczej mówiąc, aparatu zdolnego do wykonywania pewnych poleceń spełnia tu właściwie Kopciuszek — który sprząta, gotuje, pierze i wykonuje inne czynności, zlecone jej przez macochę. Macocha w tym modelu pełni rolę programisty — a zatem osoby, wykorzystującej posiadany aparat do wykonywania pewnych czynności po uprzednim ich zaprogramowaniu.

Przypuśćmy dalej, że pewnego dnia macocha „programuje" pasierbicę, ustalając jej rozkład zajęć w sposób następujący:

Upierz i rozwieś bieliznę. Wysprzątaj kuchnię. Wytrzep dywan. Ugotuj obiad i pozmywaj naczynia... i tak dalej.

Wszystkie polecenia wykonania pewnej czynności wyrażone są tutaj przy pomocy zdań rozkazujących. (Nie jest to jedyny w języku polskim sposób wyrażania rozkazu — mamy jeszcze do dyspozycji np. posłużenie się równoważnikiem zdania, np. Wstać! Koniec mycia!)

Nadajmy teraz naszemu tekstowi jednolitą formułę, przekształcając wszystkie zdania w zdania pojedyncze. Oto efekt.

Tak przystosowany tekst stanowi ciąg zdań utworzonych już według pewnych ustalonych zasad, np. każde z nich odnosi się dokładnie do jednej czynności, która ma być wykonana: upierz, rozwieś, wysprzątaj, ugotuj, pozmywaj. Nie jest to jedyna prawidłowość, za moment przejdziemy do następnych. W każdym razie ciąg zdań rozkazujących, z których każde utworzone jest w myśl pewnych ściśle określonych zasad możemy już nazwać **programem**.

Przyjrzyjmy się pojedynczemu zdaniu.

Powiedzieliśmy, że w każdym zdaniu mówimy o pewnej czynności do wykonania, np.

ROZWIEŚ bieliznę.

ŚPIJ.

Nasuwa się tutaj analogia z Logo:

FORWARD 250

HOME

Pierwsze z podanych zdań zawiera dwa wyrazy, drugie natomiast — tylko jeden. Dlaczego?

Przypuśćmy, że pewnego razu macocha nakazuje Kopciuszkowi:

KUP.

Niestety, jest to polecenie niewykonalne i dziewczę ma pełne prawo odmowy jego wykonania. Wprawdzie dokładnie wiadomo, jakie czynności należy wykonać, aby coś kupić, niemniej jednak do pełnego określenia żądanej czynności brakuje jeszcze informacji dodatkowej: co kupić należy.

Prawidłowo powiemy przecież np.

KUP gazetę.

Oto analogiczna sytuacja w Logo. Podanie np.

PRINT

spowoduje przerwanie pracy komputera i wydruk komunikatu

Not enough inputs to PRINT

(Za mało danych dla PRINT)

W Logo czynność, która ma być wykonana nazywa się **procedurą,** zaś wszelkie dodatkowe informacje potrzebne do jej zrealizowania — **parametrami**. Zlecenie wykonania pewnej czynności to **wywołanie procedury**.

Zauważmy, że procedura może mieć więcej niż jeden parametr.

IDŹ naprzód trzy kroki

Procedura IDŹ korzysta w tym przypadku z trzech parametrów. Pierwszy z nich określa kierunek przemieszczenia, drugi ilość jednostek odległości, która ma być pokonana, ostatni wreszcie określa miarę tych jednostek (kroki, mile, metry itp.)

Oto przykłady procedur Logo z trzema parametrami

(PRINT :x "Kuba [jest największym obżartuchem])

IF :a = "a [FORWARD 10] [PRINT [Nie można zrobić ruchu]]

lub, nieco przejrzyściej

(PRINT "To "jest "to!)

IF TRUE [TYPE TRUE] [TYPE FALSE]

Z drugiej strony, w niektórych przypadkach, parametry mogą nie być potrzebne w ogóle. Świadczy o tym podany już przykład:

ŚPIJ.

Czynność jet tu dostatecznie dobrze określona przez czasownik. Procedurę tego typu nazywać będziemy **bezparametrową**. W Logo istnieją również takie procedury, np.

HOME
PENUP
TEXTSCREEN
FENCE

Niektóre z powyższych nazw stanowią zlepek dwóch słów, celowo jednak istnieje obowiązek pisania ich łącznie. W Logo nazwa procedury nie może zawierać spacji — innymi słowy musi to być jeden wyraz.

Powróćmy do monologu macochy do Kopciuszka. Zakładamy, że w danym momencie dziewczyna posiada zdolność wykonywania pewnego zbioru procedur. Procedury, dla których umiejętność ta nie jest uwarunkowana ingerencją macochy-programisty, nazwiemy procedurami **pierwotnymi**. To znaczy, że w momencie rozpoczęcia „programowania" Kopciuszek już umie te czynności wykonać. Możemy sobie wyobrazić, że dziewczyna nauczyła się ich (nie interesuje nas, w jaki sposób) jeszcze przed pojawieniem się macochy w domu.

Czy procedury te stanowią jedyne czynności, które Kopciuszek może wykonać? Otóż nie. Wyobraźmy sobie, że macocha nakazuje w pewnym momencie:

PRZEWRÓĆ omlet.

pasierbica zaś nie umie wykonać tej czynności, nie wiedząc, co należy zrobić, aby omlet przewrócić. Pozwólmy macosze wyjaśnić, co kryje się pod tą nazwą.

Aby PRZEWRÓCIĆ omlet (należy)

WZIĄĆ talerz.

PRZYKRYĆ patelnię talerzem.

ODWRÓCIĆ patelnię talerz.

ZSUNĄĆ omlet na patelnię.

W ten sposób czynność została dobrze określona jako złożenie innych, już znanych czynności. W ostatnim zdaniu potrzebne jest wyjaśnienie, dlaczego użyliśmy kropki w celu połączenia dwóch wyrazów. Przyimek „na" nie stanowi osobnego parametru, a jedynie wskazuje na miejsce, stanowiące cel przemieszczenia — patelnię. Procedura ZSUNĄĆ ma zatem dwa parametry.

Mówimy, że nowa procedura PRZEWRÓCIĆ została **zdefiniowana**. Jeżeli tylko sierotka zapamięta daną definicję — a przypuszczamy, że pamięć ma dobrą — nie powinno być więcej kłopotów z przewracaniem omletu czy naleśnika. (Tymczasem zapewne ten pierwszy się spalił).

Warto uczynić spostrzeżenie, że zdefiniowawszy przy pomocy procedur pierwotnych inne procedury możemy z nich z kolei składać następne, itd. Mamy zatem możliwość **programowania strukturalnego**.

Jak wygląda definiowanie procedur w Logo?

Mamy na początku pewien zestaw procedur pierwotnych, jak choćby:

FORWARD
LEFT
PRINT
REPEAT
IF
...

Chcąc zdefiniować nową procedurę, piszemy np.

TO KWADRAT :bok
HIDETURTLE
REPEAT 4 [FORWARD :bok LEFT 90]
SHOWTURTLE
END

Słowo END jet tu jedynie znacznikiem końca definicji i nie ma znaczenia dla jej treści. Programując dalej, możemy już uzyskać program o strukturze złożonej, używając KWADRAT w definicjach innych procedur, np.

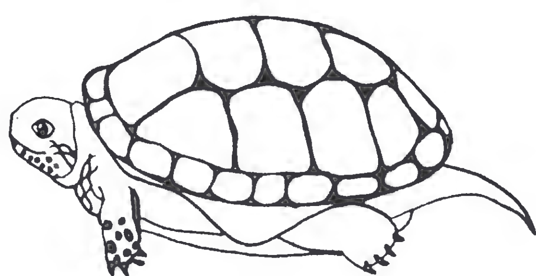
TO SZACHOWNICA :bok
REPEAT 4 [KWADRAT :bok/ 2 RIGHT 90]
END

Tryb definiowania procedur jest tutaj podobny, jak w przypadku macochy i Kopciuszka. Różnice sprowadzają się do kwestii gramatycznych i interpunkcyjnych. Na przykład istnienie słowa END na końcu definicji Logo. Trzeba też mieć na uwadze, że Logo nie zna odmiany czasowników i rzeczowników oraz że zdania w Logo są oddzielane spacją, a nie np. kropką, jak w języku naturalnym.

Marcin Waligórski (lat 20)

KOPCIUSZEK

- Upierz bieliznę
- Rozwieś bieliznę
- Wysprzątaj kuchnię
- Wytrzep dywan
- Ugotuj obiad
- Pozmywaj naczynia



Historia ta dzieje się w bardzo, bardzo dawnych czasach, kiedy to świat cały pełen był małych hobbitów, trochę większych krasnoludów, ogromnych smoków oraz innych przedziwnych stworów.

Wszystko było wówczas o wiele lepsze, czarodziejskie i zupełnie inne. Zupełnie inne pod każdym względem prócz jednego: czar złota bowiem, nie prysnął do dzisiaj. W tej historii złota jest bardzo dużo, cała góra. Na dodatek można zdobyć je szalenie łatwo: wystarczy tylko parę razy umknąć śmierci a na koniec zakłuć wielkiego, ziejącego ogniem smoka. Jakie to proste, prawda? Mały hobbit też tak pewnie myślał, gdy pakował się w tę awanturę. A było to tak:

„W pewnej norze ziemnej mieszkał sobie pewien hobbit. Nie była to szkaradna, brudna, wilgotna, nora, rojąca się od robaków i cuchnąca błotem, ani też sucha, naga, piaszczysta nora bez stołka, na którym by można usiąść, i bez dobrze zaopatrzonej spiżarni; była to nora hobbita, a to znaczy: nora z wygodami". (Tak właśnie zaczyna się książka J.R.R. Tolkiena „Hobbit, czyli tam i z powrotem" wyd. polskie ISKRY, 1985, przetłumaczona przez p. Marię Skibniewską).

W tej właśnie „norze z wygodami", pewnego dnia, zjawili się nieoczekiwani goście: trzynastu dość osobliwych krasnoludów (nie mylić z krasnoludkami, które są rodem z innych bajek), oraz Gandalf, wędrowny czarodziej. Gospodarz nory, nasz hobbit, nazywał się Bilbo Baggins i był ponoć niezwykle spokojnym hobbitem. Jeśli ktoś przypadkiem nie wie jeszcze co to jest hobbit, to w paru słowach wyjaśniam: małe, z wypiętym brzuszkiem, ubiera się kolorowo, wesołe, ma kudłate łapy a apetyt odwrotnie proporcjonalny do swojego wzrostu — ponoć zjada dwa obiady dziennie! (Choć, mówiąc prawdę, znam kilku takich co odpowiadają temu rysopisowi a wcale nie są hobbitami, wręcz przeciwnie). Ale wróćmy do gości Bagginsa. Wśród ucztowania, śpiewów i muzyki zaproponowali oni małemu hobbitowi przeżycie niezwykłej przygody: udział w wyjątkowo niebezpiecznej i dalekiej wyprawie po złoto.

Jeden z krasnoludów, Thorin, syn Thraina a wnuk Throra, Króla spod Góry opowiedział taką oto historię:

Przed laty, wyparta z dalekiej północy gromada krasnoludów dotarła do stóp Góry i to właśnie miejsce wybrali oni sobie na nową siedzibę. Wykuli więc w niej długie tunele, ogromne podziemne hale i warsztaty a przy okazji znaleźli (drobnostka!) furę złota i drogich kamieni. Bogactwo to, przysporzyło im sławy i szacunku wśród ludzi zamieszkujących okolice. Był to niewątpliwie złoty okres dla krasnoludów, którzy żyjąc w dostatku i wygodzie mogli spokojnie hodować swoje brody.

Lecz jak to zwykle w bajce bywa (w życiu niestety często też) do akcji wkroczył czarny charakter — w tym przypadku straszny smok o imieniu Smaug. Ten nie zadowalał się jedynie pożeraniem owieczek i baranków oraz nieszczęsnych dziewic ale uwielbiał wprost okradać ze złota i klejnotów ludzi, elfy i krasnoludy. Chciwy i zły Smaug, gdy tylko usłyszał o skarbach, przyfrunął, spadł na Górę, zionął ogniem tak, że okoliczne lasy stanęły w płomieniach i zabrał się do zabijania właścicieli skarbów. Tylko nielicznym udało się uciec, wśród nich był i Thorin. Smok zgromadził wszystkie bogactwa w najgłębszej jaskini, usypał z nich górę a na jej szczycie zrobił sobie legowisko, którego prawie nie opuszczał, a jeśli, to tylko po to, żeby co nieco przekąsić — czego efektem jednak było zupełne spustoszenie okolicy.

I tak, bogata dotąd i hojna dla ludzi i krasnoludów kraina stała się teraz dzika i groźna. Ci, którzy ocaleli, przez wiele lat błąkali się po świecie, cierpieli niedostatek i biedę. Nie wyrzekli się jednak zamiaru odzyskania swoich skarbów, a tym bardziej dokonania zemsty na straszliwym Smaugu.

Tak więc, znana nam już grupa trzynastu krasnoludów, na czele której stał Thorin, miała zamiar spełnić słowa starej przepowiedni, że potomek Throra i Thraina wróci kiedyś, a wówczas z Góry spłynie złoto i cały kraj znowu stanie się radosny i bogaty.



HOBBIT

W organizowaniu wyprawy, krasnoludom pomagał Gandalf. On to właśnie wybrał naszego małego hobbita jako czternastego uczestnika wyprawy i chyba tylko on jeden wiedział jak wiele przeszkód stanie im na drodze: złe gobliny, odrażające, a co gorsza potwornie głodne pająki to tylko mała część tego co miało ich spotkać.

W tej grze Ty jesteś hobbitem (co wcale nie znaczy, że masz zjadać 2 obiady dziennie!) a Twoim zadaniem jest pomóc krasnoludom w odzyskaniu złota. A gdy złoto będzie już wasze musisz zabrać swoją część, powrócić z nią do norki i schować w skrzyni. Ale zanim to się stanie, czeka Cię niezwykle daleka, pełna niebezpieczeństw ale jednocześnie fascynująca wędrówka, która będzie wymagała od Ciebie wyjątkowo dużo sprytu i zręczności. Na szczęście nie będziesz sam. Spotkasz wielu przyjaciół, na których pomoc zawsze możesz liczyć. A może i oni będą potrzebowali Twojej pomocy? Musisz dbać zwłaszcza o Thorina, bo jeśli on zginie to i Ty masz marną szansę przeżycia.

A TERAZ PARĘ CENNYCH INFORMACJI I RAD

- **Podczas, gdy rozmawiasz z komputerem i bohaterami bajki posługujesz się językiem INGLISH (o zasadach jego stosowania dowiesz się dokładniej w artykule INGLISH na stronie 12).**
- **Jeśli potrzebujesz więcej czasu na odczytanie informacji z ekranu trzymaj wciśnięty jakikolwiek klawisz (chyba warto, bo komputer, choć nie pytany, udziela często ciekawych informacji).**
- **Gdy po raz pierwszy zjawiasz się w jakimś miejscu, komputer podaje Ci dokładny opis, często jest to grafika. Mówi Ci gdzie jesteś i jak to miejsce wygląda oraz podaje w jakich kierunkach możesz udać się.**
- **Gdy po raz drugi znajdziesz się w tym samym miejscu, komputer podaje Ci jedynie krótki opis, a jeśli chcesz uzyskać ponownie pełną informację na temat tego miejsca to musisz popatrzeć wokół.**
- **Podczas swojej wędrówki znajdziesz wiele dziwnych i niezwykłych przedmiotów. Niektóre z nich będą miały czarodziejską moc, inne będą służyły jako broń, a możesz znaleźć także coś do zjedzenia — więc nie zapomnij zabrać tego wszystkiego ze sobą.**
- **Mimo, że wędrujesz po zaczarowanej krainie to jednak i tu obowiązują podstawowe prawa fizyki:**
  — nie podniesiesz zbyt ciężkiej dla Ciebie rzeczy (ale pamiętaj, że masz większych i silniejszych przyjaciół!)
  — nie włożysz większego przedmiotu w mniejszy
  — nie możesz nic wyjąć ani włożyć do pojemnika, który jest zamknięty
  — nie możesz przenikać przez zamknięte drzwi niczym duch (musisz zawsze otworzyć je odpowiednim kluczem)
  — walka uczyni cię słabszym, ponieważ wymaga wiele energii. Musisz zatem odżywiać się regularnie, aby odzyskać siły. (Ale ostrzegamy — nie bądź żarłokiem!).
- **Masz do dyspozycji aż 8 kierunków geograficznych: NORTH (Pn.), SOUTH (Pd.), EAST (Wsch.), WEST (Zach.), NORTH-EAST (Pn--Wsch.), NORTH-WEST (Pn-Zach.), SOUTH--EAST (Pd-Wsch.), SOUTH-WEST (Pd-Zach.), oraz UP (w górę) i DOWN (w dół).**
- **Możesz przechodzić przez otwarte już okna i drzwi a także wyglądać przez nie, jeśli chcesz wiedzieć co cię za nimi może spotkać.**
- **Jeśli ktoś, z kim chcesz porozmawiać, odszedł możesz iść za nim (np. FOLLOW GANDALF).**
- **Przechodzenie przez rzeki, parowy i przepaści może być trudne i niebezpieczne. Czasami możesz je przeskoczyć a czasem będziesz musiał użyć czegoś co wcześniej znalazłeś.**
- **Wszystkie postacie, które biorą udział w tej przygodzie działają niezależnie od tego co robisz Ty, mają swoje własne przygody i pomysły.**
- **Nie mów na raz zbyt dużo do jednej osoby, bo pomyśli ona, że jesteś nudziarzem i nie będzie chciała Ci pomóc. (Z tego samego powodu staraj się rzadko używać słowa ALL — wszystko).**
- **Czas płynie i gdy zwlekasz z wykonaniem jakiejś czynności może zaistnieć zupełnie nieprzewidziana dla Ciebie sytuacja, ponieważ inne postacie w tym samym czasie coś przecież robią.**
- **W tej grze jeden problem można rozwiązać na wiele sposobów i dlatego też, gdy zaczynasz grę od nowa możesz spotkać się z zupełnie nowymi sytuacjami.**

A WIĘC ZA KAŻDYM RAZEM MASZ SZANSĘ PRZEŻYĆ ZUPEŁNIE NOWĄ PRZYGODĘ!!!

Hobbitowi przejście tam i z powrotem (czyli wycieczka na trasie: nora — jaskinia smoka — nora) zajęło bez mała rok. Co prawda miał mapę, którą na samym początku wyprawy podarował mu Gandalf, lecz nie na wiele mu się ona przydała. Mapa, którą Tobie dajemy, choć daleko jej do doskonałości, przyda Ci się na pewno. A już z pewnością nie pozwoli Ci zabłądzić i pomoże w odnalezieniu właściwej drogi.

A jeśli dodatkowo uda Ci się zdobyć książkę J.R.R. Tolkiena, na podstawie której Philip Mitchel, Veronika Megler, Alfred Milgrom i Stuart Ritchie stworzyli swoją komputerową bajkę, to najpierw przeczytaj ją bardzo uważnie, a dopiero potem wraz z krasnoludami...

Ponad gór omglony szczyt
Lećcie, zanim wstanie świt,
By jaskiniom, lochom, grotom
Czarodziejskie wydrzeć złoto!

Teresa Brojak

zagraj z Bajtkiem

hobbit czyli tam i z powrotem hobbit czyli tam i z powrotem hobbit czyli

Mapa

1. IDŹ NA PÓŁNOC OD DOMU BILBO ... ABY DOJŚĆ DO WIELKIEJ RZEKI
2. IDŹ NA WSCHÓD OD LEŚNEJ BRAMY ...

DARK STUFFY PASSAGE

D.S.PASSAGE

KORYTARZ

CIEMNY KORYTARZ

D.S.PASSAGE

D.S.PASSAGE

D.S.PASSAGE

D.S.P.

D.S.P.

D.S.R.

D.S.R.

hobbit

hobbit czyli tam i z powrotem hobbit czyli tam i z powrotem hobbit czyli



tam i z powrotem hobbit czyli tam i z powrotem hobbit czyli tam i z powrotem
hobbit czyli tam
BEWITCHED GLOOMY PLACE - ZACZAROWANE MROCZNE MIEJSCE
SKARB
N
W
E
S
DRAGONS DESOLATION
CELLAR-PIWNICA
CIEMNY
KORYTARZ
DARK STUFFY PASS-AGE
tam i z powrotem hobbit czyli tam i z powrotem hobbit czyli tam i z powrotem



Bajtkowa lista przebojów

Dziesiątki tysięcy komputerów, tysiące gier komputerowych, które z tych programów są naprawdę dobre? Grałeś już setki razy, jedne wyrzucałeś po kilku minutach, inne skradły Ci całe dni i grasz w nie nadal — irytują Cię, czasami już masz dosyć i... grasz — może właśnie Ty wiesz, które z nich są najlepsze! Wybierz 10 z nich i napisz do nas. Proponujemy startową dziesiątkę (Twoje typy mogą być całkiem inne).

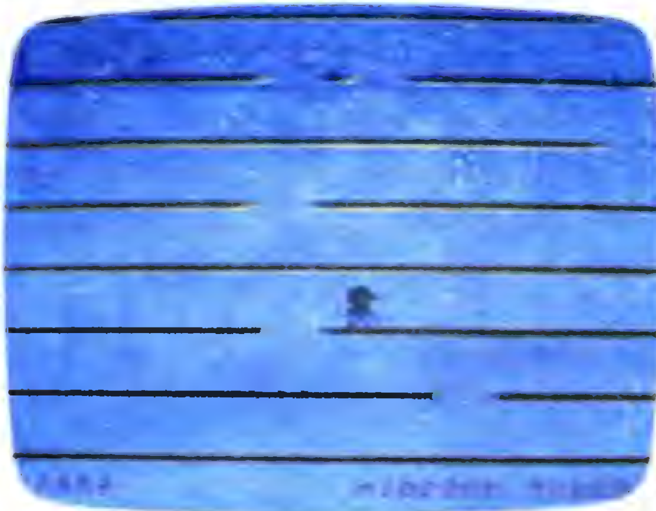
1. JUMPING JACK
2. THE WAY OF THE EXPLODING FIST
3. TAPPER
4. TIR NA NOG
5. ATIC — ATAK
6. SABRE WULF
7. PSYTRON
8. HOBBIT
9. GHOSTBUSTERS
10. KNIGHT LORE

Za miesiąc pierwsze notowania listy. Jeżeli trafisz w dziesiątkę nagroda jest Twoja, jeżeli będziesz jedyny — nagroda będzie jedyna w swoim rodzaju. Musisz spróbować — na pewno warto! Czekamy na Twoje propozycje do Bajtkowej Listy Przebojów.
Dodatkowa premia za opisy typowanych gier.

Sławek

Nasz adres:

BAJTEK
ul. Wspólna 61
00-687 Warszawa
LISTA PRZEBOJÓW



„Szybki i śmiały jest Jumping Jack Jego historia odsłoni się wnet..."

JUMPING JACK
Imagine Software

Jumping Jack to gra, której brakuje praktycznie wszystkiego, co — na pozór — decyduje o atrakcyjności. Prosta, wręcz uboga grafika, nieskomplikowane efekty dźwiękowe, strategia nie zmuszająca do specjalnego wysiłku intelektualnego. Brak zaskakujących pomysłów.

Zawsze w górę! W tych dwóch słowach można zamknąć cel całej zabawy. Zadanie polega na wskakiwaniu na coraz to wyższe piętra galerii. Trzeba jedynie uważać, by nie wpaść w dziurę pojawiającą się pod stopami. Co prawda nie jest to takie proste, zwłaszcza w dalszych etapach gry, kiedy to śmiały odkrywca Jack musi zmykać przed ścigającymi go w coraz to większej liczbie parowozami, autobusami itp.

Dlaczego więc ta prosta — nieżyczliwi mogliby powiedzieć prymitywna — gra plasuje się na czołowych miejscach list przebojów gier komputerowych? (Nie tylko w BAJTKU). Dlaczego oceniana jest wyżej niż wiele innych, pozornie bardziej efektownych? Na te pytania nie da się odpowiedzieć, nim nie zawrze się osobistej znajomości ze sławnym podróżnikiem o imieniu Jack.

Może znudziło nam się ciągłe strzelanie do statków kosmicznych, zabijanie potworów i demonów, zatapianie okrętów. Warto czasem poskakać sobie dla utrzymania kondycji. A może tajemnica tkwi w balladzie o Jumping Jacku, której kolejne fragmenty komputer opowiada nam w nagrodę za przejście każdego etapu gry. Swobodnym tłumaczeniem początku tej fascynującej opowieści rozpoczęliśmy naszą mikrorecenzję. Wszystkich ciekawych ciągu dalszego zapraszamy do gry.

(rp)

TAPPER
Us Gold

Czy zastanawiałeś się kiedyś jak trudno być dobrym barmanem? Jeśli jeszcze nie przyszło ci to do głowy spróbuj czy potrafisz.

W grze Tapper, opracowanej przez firmę US GOLD, obsługujesz gości w wyjątkowo ruchliwej knajpie. Najgorsze jest to, że klientela to same typy spod ciemnej gwiazdy: kowboje, punkowcy, sympatyczne na pierwszy rzut oka, choć nie mniej agresywne od reszty towarzystwa panienki etc. Wszyscy mają jedno życzenie — Pepsi!

Nie można powiedzieć, że grzeszą zbytnią cierpliwością, jeśli tylko któryś z nich dojdzie do końca kontuaru i nie dostanie swojej szklaneczki, łapie cię za kołnierz i teraz ty sam jedziesz po gładkim blacie w charakterze chłodzącego napoju. Twój pracodawca nie toleruje takich incydentów. Nie lubi także gdy tłuką się jego szklanki. Musisz więc uważać, by łapać wszystkie puste naczynia i nie posyłać więcej napojów niż jest chętnych. Pięć takich wpadek powoduje wyrzucenie z pracy i koniec gry. Chyba, że pracowałeś wyjątkowo dobrze. Szef może — w drodze wyjątku — zapomnieć o jakimś przewinieniu.



To jeszcze nie wszystko. Co pewien czas do baru wpada pewien szczególnie niesympatyczny facet w masce na twarzy. Do wszystkich — z wyjątkiem jednej — puszek pepsi dosypuje truciznę. Potem przestawia puszki. Ty musisz wskazać bezbłędnie tę, której bandyta nie miał w rękach.

Pod względem grafiki i animacji gra opracowana jest raczej starannie. Nie to jednak decyduje o jej atrakcyjności. Najważniejsze w tym przypadku, było poczucie humoru programistów i niekonwencjonalny pomysł.

(rp)



THE WAY OF THE EXPLODING FIST
Melbourne House

Cały świat pasjonował się przygodami legendarnego już Bruce'a Lee. Nie każdego z nas stać jednak na wielogodzinne treningi, tym bardziej, że firma Melbourne House daje możliwość osiągnięcia mistrzowskich umiejętności i najwyższych stopni wtajemniczenia bez potrzeby ruszania się z wygodnego fotela.

„The way of the exploding fist" („Droga eksplodującej pięści") to gra reklamowana jako „najlepszy z możliwych sposobów wciągnięcia w walkę twoich przyjaciół".

Uczestniczyć w niej może jedna lub dwie osoby. W pierwszym przypadku przeciwnikiem jest postać wykreowana przez komputer. Początkowo zwyciężasz stosunkowo łatwo, ale w miarę rozwoju gry ten drugi staje się coraz bardziej bystry i zwinny.

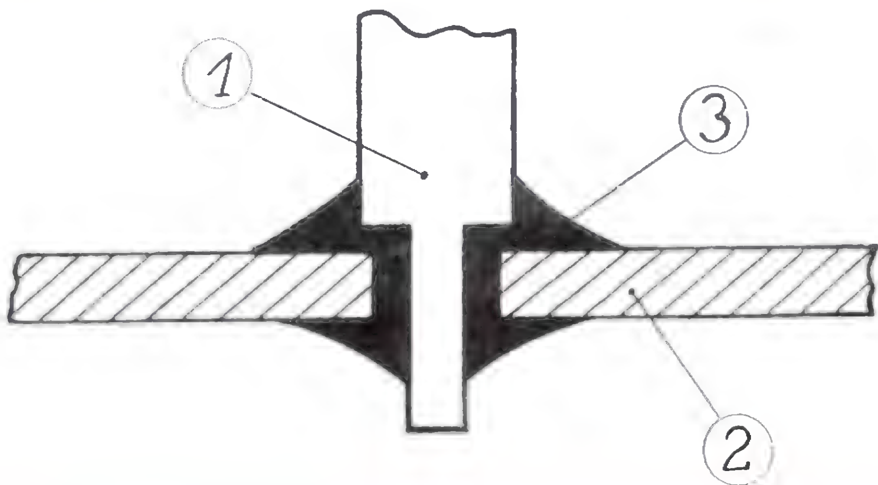
Swoim zawodnikiem sterujesz przy pomocy drążka sterowego lub też poprzez klawiaturę. Daje to możliwość uzyskania 16 pozycji. Zapamiętanie wszystkich ruchów nie jest łatwe; rzecz polega na tym, że poruszając drążkiem powodujemy wędrówkę postaci na ekranie, a używając dodatkowego przycisku zadajemy różne ciosy.

Najlepiej zacząć od wersji gry 2-osobowej, walcząc z nieruchomym przeciwnikiem. Gdy już osiągniemy pewną wprawę nadszedł czas, aby przyjąć wyzwanie komputera. Na koniec można zaprosić do gry przyjaciela.

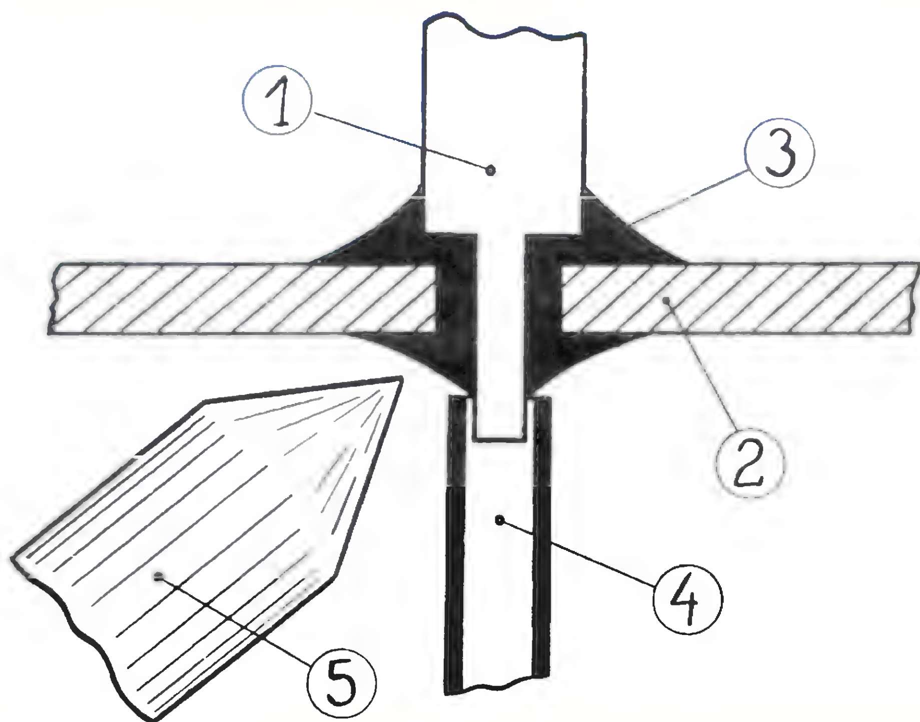
Pierwsza wersja „Eksplodującej pięści" opracowana została na Commodore, a następnie adaptowana na Spectrum i Amstrada. Zwraca uwagę starannie opracowana grafika. Ruchy postaci są spokojne i dokładne. Jest to duży postęp w stosunku do „Kung-fu" (starszej wersji tego programu) i za to firmie Melbourne House należą się brawa

(spo)

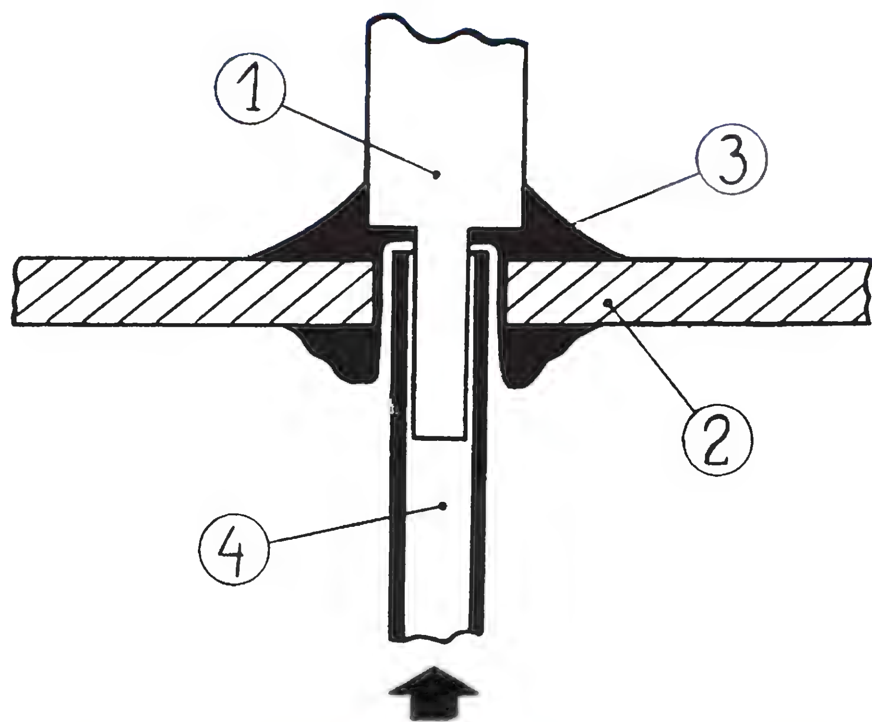
JAK NAPRAWIĆ KOMPUTER?



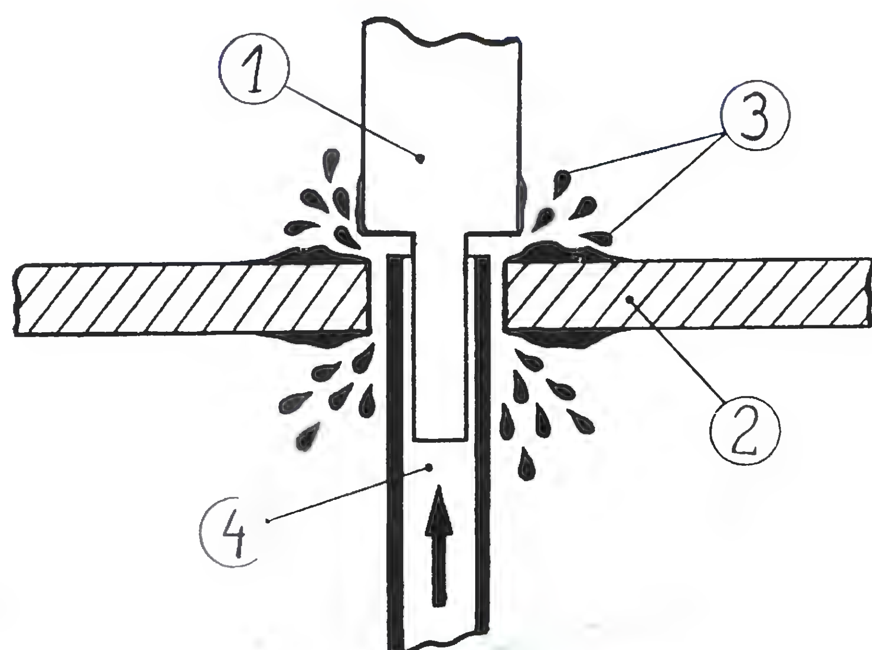
Rys. A



Rys. B



Rys. C



Rys. D

Każdy chyba elektronik-amator styka się czasem z problemem wylutowania wielonóżkowego układu scalonego. Niestety, układy takie nie zawsze są montowane w podstawkach, co powoduje, że ich wyjęcie z płytki bez uszkodzenia ścieżek lub samego układu jest prawie niemożliwe. Nie ma jednak sytuacji bez wyjścia. Niżej opisany sposób pozwala na bezpieczne wymontowywanie układów o dowolnej liczbie nóżek, nawet obustronnie wlutowanych w płytkę. Potrzebna jest do tego celu strzykawka z igłą, lutownica o małej mocy (15–25 W), cienkie szczypce oraz... ktoś do pomocy.

Przygotowanie narzędzi

Najważniejszym narzędziem jest odpowiednio przygotowana igła. Najpierw musimy znaleźć igłę o takiej średnicy wewnętrznej i zewnętrznej, aby igła mieszcząc się w otworze płytki drukowanej dała się jednocześnie nasunąć na wystającą część nóżki układu scalonego. Praktyka wykazała, że wymaganiom tym odpowiada oszlifowana igła nr 9. Igłę przystosowujemy do pracy odcinając pilnikiem jej ostre zakończenie (jak na rys. E). Strzykawka może być dowolna (autorzy artykułu używali strzykawki 10 cm^3). Następnym krokiem jest znalezienie szczypców o takich rozmiarach, aby można nimi było uchwycić nóżkę wychodzącą z układu (celem odprowadzenia ciepła). Trzecie narzędzie to lutownica. Stanowczo odradzam tu lutownice transformatorowe ze względu na płynące w jej grocie prądy i silne pole elektromagnetyczne. Grot lutownicy powinien być tak spiłowany, aby przy lutowaniu dotykać tylko jednego punktu lutowniczego, ale za to na możliwie dużej powierzchni. Należy sprawdzić, czy grot lutownicy — w wyniku przebicia — nie znajduje się pod napięciem.

Sposób naprawy

Idea pomysłu polega na wydmuchiwaniu cyny z otworów w płytce. Żaden inny sposób przy tak małych wymiarach punktów lutowniczych nie zapewnia dobrego oczyszczenia połączenia z cyny. Przystępując do pracy należy najpierw umocować płytkę (odradzam imadło) i dobrze ją oświetlić. Kolejne fazy wylutowania pokazane są na rysunkach A, B, C i D.

Na rys. A widzimy sposób w jaki układy scalone są montowane w płytce. Rys. B pokazuje, jak należy nałożyć igłę na końcówkę układu scalonego. Po nałożeniu igły (przypominam, że druga osoba musi w tym czasie trzymać szczypcami nóżkę układu po drugiej stronie płytki) należy przyłożyć do połączenia dobrze nagrzaną lutownicę. Gdy cyna roztopi się, należy wcisnąć igłę w otwór w płytce, cofnąć lutownicę i natychmiast nacisnąć tłoczek strzykawki. Cyna powinna wyprysnąć z drugiej strony (uwaga na oczy osoby trzymającej szczypce!). Te fazy pokazane są na rysunkach B i C. Podobną operację należy przeprowadzić z każdą końcówką układu. Gdy wylutujemy już wszystkie nóżki, należy delikatnie spróbować układ wyjąć. Na ogół nie udaje się to za pierwszym razem, niektóre nóżki trzeba wylutowywać kilkakrotnie.

Podczas pracy należy zwrócić uwagę na nagrzewanie układu scalonego — nieraz trzeba zrobić przerwę na ostygnięcie. Należy również dosyć delikatnie wycofywać igłę z otworu w płytce — zastygająca cyna nierzadko łączy igłę z nóżką układu i nieostrożnym ruchem można po prostu urwać końcówkę układu. Gdy już wyjmiemy układ z płytki, pozostaje jeszcze oczyszczenie go z cyny oraz sprawdzenie, czy wytryskująca z otworów cyna nie spowodowała zwarcia gdzieś na płytce. Jeśli jednak przystąpimy do pracy z dużym zapasem cierpliwości (wylutowanie układu 40-nóżkowego trwa 2–3 godziny) i uwagi, możemy być pewni sukcesu. Powodzenia!

Michał Silski
Dariusz Rotsztyn
Klaudiusz Dybowski

Rys. E



1. nóżka układu scalonego 2. płytka (obw. drukowany) 3. cyna 4. igła 5. grot lutownicy

Jesienią 1983 roku pojawiły się w Stanach Zjednoczonych nowe komputery firmy Atari 600XL i 800XL. Był to bardzo trudny okres dla producentów mikrokomputerów. Bankructwo Osborne, wycofanie się Texas Instruments z TI-99/4A, i kłopoty IBM z PC Junior to przykłady przejściowego załamania się rynku. Dotyczyło to także Atari. Firma znalazła się w finansowych kłopotach co doprowadziło do znacznego okrojenia śmiałych zamierzeń. Produkowane od 1979 roku modele 400 i 800 planowano zastąpić serią XL: 600 XL, 800XL, 1200XL, 1400XL, 1450XLD (wbudowana stacja dysków) i 1600XL. Praktycznie w produkcji znalazły się tylko 600XL i 800XL.

Atari Inc. była częścią dużego koncernu Warner Communication Company, który chylił się ku upadkowi m.in. z powodu nakręcenia kilku niekasowych filmów.

Tymczasem latem 1984 roku doszło do ostrego konfliktu wewnątrz firmy Commodore, której szefem i współwłaścicielem był Jack Tramiel, polski emigrant, ojciec słynnego C64 rozstał się on z Commodore i wykupił podupadającą wtedy Atari. Wraz z nim odeszło wielu konstruktorów oraz ludzi zajmujących się oprogramowaniem i marketingiem. „Chcę zgnieść Commodore" powiedział na pierwszym spotkaniu z dziennikarzami nowy szef Atari. Była to wtedy trochę czcza zapowiedź — dzisiaj wiemy, że nie bez podstaw, bo znamy już komputer roku 1985, którym został Atari 520ST. W najbliższym czasie mamy zamiar przedstawić opis oraz pierwsze wrażenia z użytkowania tego komputera.

520ST jest dzisiaj motorem rozwoju Atari co nie znaczy, że zapomniano o amatorach tańszych urządzeń i użytkownikach komputerów 8-mio bitowych. W 1985 roku szaloną popularność zdobył 800XL wraz z młodszym, a raczej mniejszym bratem 600XL. Był to najlepiej sprzedawany komputer domowy w Wielkiej Brytanii (wg PC World nr 6) i w swojej klasie wygrywał wszystkie konkursy popularności. Anglia jest drugim, po USA, krajem, gdzie Atari jest czołowym komputerem. Jack Tramiel zapowiedział kontynuację kompatybilnych z XL komputerów noszących nazwę XE, z których 130XE posiada 128 KB RAM-u i jest już znany polskim użytkownikom.

ATARI 800XL

Opis modelu 800XL rozpoczniemy od parametrów technicznych:

Wymiary: 54 × 378 × 197 mm
Mikroprocesor: 6502C 1.79 MHz
Pamięć: 64 kB RAM
24 kB ROM
w tym 8 kB Basic
16 kB system operacyjny.
Klawiatura: typ QWERTY, 62 pojedyncze klawisze dla poszczególnych znaków oraz rozkazów operatora w tym 5 klawiszy pomocniczych, z tego 4 programowalne.
Grafika: rozdzielczość 320 × 192 punkty, 256 barw, 24 wiersze po 40 znaków.
Dźwięk: 4 niezależne kanały. W przypadku korzystania z wszystkich 4 kanałów zakres 3.5 oktawy.

Nad klawiaturą znajduje się gniazdo do przyłączania dodatkowej pamięci ROM. Użytkownik oczywiście nie ma możliwości zmiany zawartości tej pamięci. Cartridge z grą kosztuje 15-20 dolarów czyli 10% ceny stacji dysków, jednak ogromną ich zaletą jest trwałość i niezawodność. Bardzo ciekawie rozwiązano samo gniazdo — bezpośrednio można podłączyć do 16 kB, natomiast dzięki zastosowaniu przełączania banków pamięci istnieje możliwość przyłączenia 2 MB pamięci. Jest to możliwość potencjalna, praktycznie nie ma potrzeby stosowania tak dużych pamięci ROM, poza tym spowodowałoby to bardzo wolną pracę komputera.

Tylna ścianka komputera 800XL zawiera szereg gniazd wejścia/wyjścia. Są to: złącze interface'u szeregowego, wyjście modulatora TV, wyjście monitorowe, gniazdo zasilania oraz złącze równoległe.

Złącze interface'u szeregowego przeznaczone jest do komunikacji ze stacjami dysków (maksyalnie 4), magnetofonem, drukarką i modemami. Maksymalna szybkość transmisji 19200 bodów stosowana jest do dysków. Dla porównania stacja dysków Commodore 1541 — szybkość transmisji 3800 badów. Głównym typem pamięci zewnętrznej Atari z założenia były stacje dysków i stąd wynikają te doskonałe rezultaty.



FOT. LEOPOLD DZIKOWSKI

Wyjście RF — wyjście modulatora telewizyjnego służące do podłączenia odbiornika TV. Komputery sprzedawane w Europie (także te z Pewexu) wytwarzają sygnał wizyjny standardu PAL z częstotliwością różnicową fonii 5.5 MHz. Wynika stąd konieczność przestrajania fonii komputera w przypadku korzystania z telewizora posiadającego tylko częstotliwość różnicową 6.5 MHz (polski standard). Wyjście monitorowe służy do współpracy z monitorem monochromatycznym lub kolorowym. Zastosowany standard jest odpowiednikiem wyjść i wejść magnetowidowych. Nie trzeba chyba tłumaczyć, jak dużą poprawę jakości obrazu uzyskuje się w przypadku połączenia komputera z telewizorem bezpośrednio omijając modulator i demodulator.

Złącze równoległe — stanowi wyprowadzenie szyn danych, adresowej i sterującej. Jest to 50 stykowe złącze, które umożliwia dostęp do szyny systemowej mikrokomputera. W jej skład wchodzą m.in. 16 linii adresowych, 8 lini danych, linie sygnałów zapisu i odczytu oraz zegara systemowego.

Z prawej strony konsoli znajdują się dwa jednakowe gniazda służące do podłączenia drążków sterowych lub manipulatorów analogowych, a także pióra świetlnego lub dodatkowej klawiatury numerycznej.

Opracowano bardzo proste sposoby rozszerzania pamięci RAM do 256 kB oraz zmianę trybu tekstowego do 80 kolumn, co czyni z domowego komputera urządzenie pomocne w profesjonalnym zastosowaniu.

Atari 800XL posiada doskonale rozbudowane możliwości graficzne i dźwiękowe dzięki trzem specjalnie zaprojektowanym układom scalonym.

STACJA DYSKÓW ELASTYCZNYCH

Znane jest zapewne wszystkim powiedzenie, że komputer bez oprogramowania to złom. Proponuję rozszerzyć to stwierdzenie o dwa nowe zdania: oprogramowanie bez opisu to ból głowy, a komputer bez stacji dysków to tylko połowa komputera.

Atari 800XL jest komputerem projektowanym z myślą o podłączeniu 4 stacji dysków elastycznych. Firma Atari proponuje jeden model 1050, osiągalny także w Pewexie. Jest to stacja na dyskietki 5.25 cala z jednostronnym zapisem i kontrolerem pozwalającym zapisywać 130 kilobajty. Jest to nowocześnie zaprojektowane urządzenie, z szybką — jak na tę klasę urządzeń — transmisją danych.

Ze stacją dysków wiąże się nierozerwalnie pojęcie dyskowego systemu operacyjnego. Najnowszy i najlepszy proponowany przez Atari-DOS 2.5 ma bardzo rozbudowane funkcje, takie jak w profesjonalnych systemach mikrokomputerowych.

Kilkanaście firm zachodnich sprzedaje swoje stacje przeznaczone dla Atari. Do najciekawszych należą: Astra, Indus, Rana, Trak i Percom. Astra BIG D jest czterogłowicową stacją dysków z dwoma kieszeniami. Firma opracowała swój własny system operacyjny, kompatybilny także z 1050, który pozwala zapisywać na jednej stronie dyskietki 360 kilobajtów, a biorąc pod uwagę, że możemy korzystać z dwóch dwustronnych dyskietek mamy bezpośredni dostęp do pamięci zewnętrznej 1440 kilobajtów. Szybkość przesyłania danych wzrasta czterokrotnie osiągając blisko 80 kilobajtów na sekundę. SPARTADOS pozwala na wielopoziomowe menu dyskietki (struktura drzewa) oraz automatycznie zapisuje datę i czas wpisywania nowego zbioru na dyskietkę.

MAGNETOFON KASETOWY

Pamięć kasetowa jest w naszych warunkach, ze względu na koszty, najpopularniejszym typem pamięci zewnętrznej. Zdecydowanie nie polecamy magnetofonu, a jeśli już to nowy model Atari CX11 albo ulepszoną wersję starego magnetofonu 1010. Jako nośniki informacji należy używać normalnych taśm dobrych firm, nie dłuższych od C60. Niestety, wiele bardzo dobrych programów, w tym większość użytkowych, zapisana jest na dyskietkach i nie ma możliwości, ze względu na potrzebę ciągłej komunikacji, sporządzenia ich kasetowych wersji.

INNE URZĄDZENIA PERYFERYJNE

Atari produkuje kilka drukarek:
1020 — czterokolorowy printer-plotter
1025 i 1029 starsza i nowsza drukarka mozaikowa
1027 — drukarka typu „letter quality printer".
Zapowiedziane są drukarki nowej generacji przystosowane do serii XE i ST. Znana w Polsce SEIKOSHA proponuje 4 modele: 100AT, 500AT, 550AT i 700AT. Jedną z najlepszych drukarek przeznaczonych dla Atari jest E PSON RXOOAT.

Inne drukarki, nie przystosowane do bezpośredniej współpracy z Atari można przyłączyć za pomocą interface'ów. Istnieje wiele rozwiązań takich urządzeń. Uniwersalny bo zawiera złącze Centronics i 4 RS232 jest Atari 850.

Do bardzo ciekawych urządzeń dodatkowych można zaliczyć produkt firmy SWP Microcomputer Product Inc. pozwalający wykorzystać system operacyjny CP/M 2.2 i CP/M 86 oraz IBM MS DOS.

Bardzo rozpowszechnione są przystawki do 80-kolumnowej organizacji ekraju.

Dostępne są także dla Atari pióra świetlne, dodatkowa klawiatura numeryczna oraz manipulatory potencjometryczne.

OPROGRAMOWANIE

Najsłabszym punktem 800XL jest jego język — rezydujący BASIC, czyli fabrycznie montowany interpreter. Zajmuje on tylko 8 kB pamięci ROM i śmiało można powiedzieć, że spośród podobnych jest jednym z najlepszych. Tylko, że nie jest to porównanie do dobrych wzorów. Projektant mikrokomputera opartego na 8-bitowym mikroprocesorze musi zdecydować jak dobry ma być system operacyjny i BASIC. Im jedno i drugie lepsze musi zajmować więcej ograniczonej do 64 kB przestrzeni adresowej procesora.

Drugą częścią oprogramowania wewnętrznego jest system operacyjny. W Atari 800XL zajmuje on 16 kB pamięci. Sam komputer ma 64 kB pamięci RAM więc łatwo można policzyć, że niewykorzystane zostaje 24 kB RAM-u albo 16 gdy wyłączymy BASIC. Nie jest to marnotrawstwo. W Atari 800XL przewidziano możliwość wczytania nowego, własnego systemu operacyjnego wykorzystującego tylko sam hardware komputera. Jest to pole do popisu dla ambitnych programistów. Można m.in. napisać krótki ok. 6 kB system operacyjny i mieć 58 kB RAM-u na Pascala. Można też w prosty sposób zastosować 800XL jako inteligentną końcówkę terminala.

Praktycznie wszystkie z dostępnych na mikrokomputery języków programowania opracowano na Atari. Było to tym łatwiejsze, że Apple II oparty jest też na 6502 i wielu programistów przełożyło swoje programy także dla Atari. Godnym szczególnego polecenia jest macroassembler autorstwa OSS o nazwie MAC-65.

Po pokonaniu pewnej bariery ilościowej zasadnicze znaczenie ma jakość oprogramowania. Aż nieprawdopodobne wydają się najnowsze programy w porównaniu do tych sprzed 3 lat. A przecież pisane są dla tego samego urządzenia. Wyliczenie ilości barw i skali dźwięku nie jest wystarczające w przypadku Atari 800XL — to trzeba samemu zobaczyć i usłyszeć, do czego wszystkich gorąco i szczerze namawiam.

Wiesław Migut

WIEŚCI

Firma At&T wyprodukowała rejestrator obrazów wizyjnych, który umożliwia zapamiętanie standardowego obrazu video w nowych, wielkiej skali integracji, pamięciach RAM i daje możliwość modyfikacji i obróbki obrazu przez komputer.

●

Firma SONY ujawniła, że bada materiały organiczne, takie jak DNA w celu zastosowania ich do nowych, elektronicznych technologii. Badania dotyczą wykorzystania porfiryny, aktywnych części jąder chlorofilu i hemoglobiny w tworzeniu trójwymiarowych „biokości".

●

Na wystawie elektroniki japońskiej firma SONY wystawiła 3 1/2", kasowalny 50MB dysk optomagnetyczny. Na dysku można zapisywać jednocześnie informacje analogowe i cyfrowe. Można zatem obok programów i danych zapamiętywać kolorowe obrazy.

●

Sinclair Radionics, Anglia. Firma wyjawiła, że wprowadzi na rynek pamięci „waflowe" o komputera Sinclair QL, które można używać zamiast dyskietek. Pamięć ma postać kostki (wafla) o boku 2,5 cm o pojemności 1MB z szeregową organizacją i bateryjnym podtrzymaniem zawartości pamięci po odłączeniu od komputera.

●

California, hobbyści radiowi utworzyli pierwszą sieć, WESTNET, sprzęgającą drogą radiową komputery domowe na terenie stanu.

●

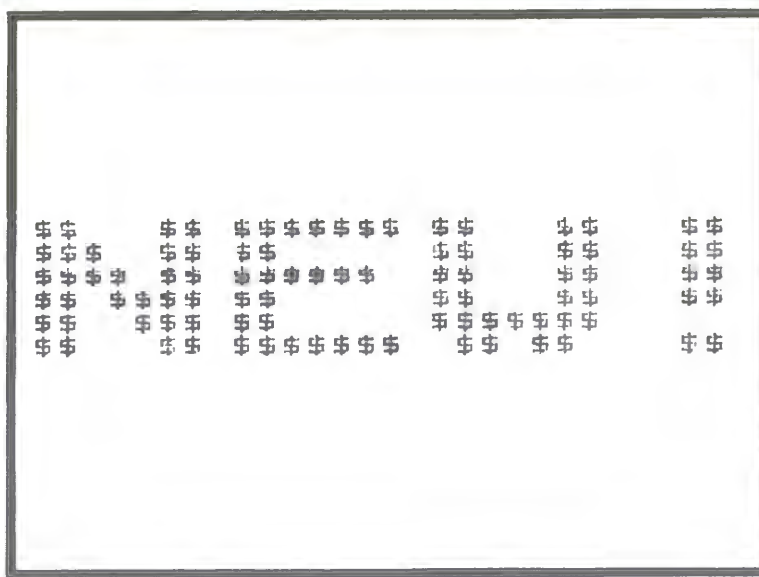
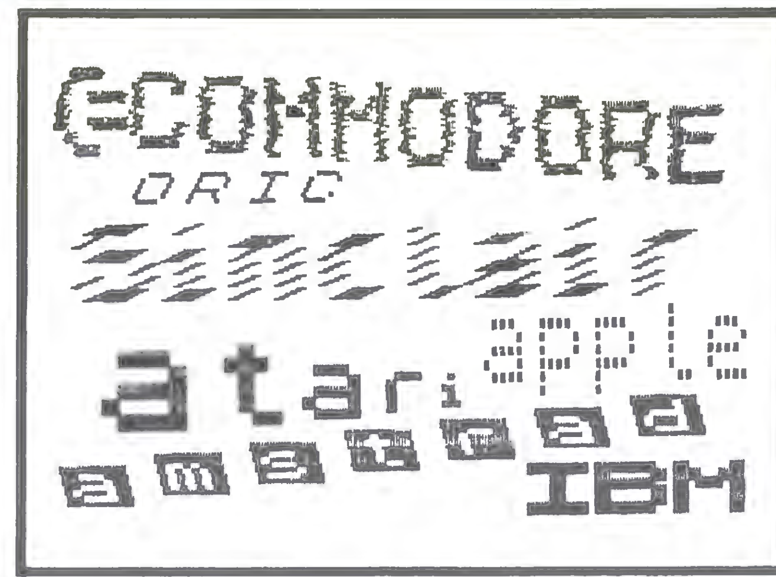
Na konferencji w Austin (USA), dotyczącej sztucznej inteligencji kilka firm (IBM, XEROX, DEC itp.) pokazało swój software. Zaprezentowano m.in. program „SCRATCHPAD II", który umożliwia przekształcenie wyrażeń algebraicznych.

●

Firma Psion, Anglia wprowadziła na rynek urządzenie, które nazywa pierwszym prawdziwym kieszonkowym komputerem na świecie. Komputerek ma wymiary 14 × 7.5 × 2.5 cm, używa dwóch wymiennych modułów EPROM po 16 kB, ma wbudowany program obsługi baz danych i 16 — to znakowy wyświetlacz. Cena ok. 140 $.

●

Jedna z najbardziej znanych na naszym rynku firm mikrokomputerowych SINCLAIR RESEARCH zrealizowała swój najnowszy projekt. W sprzedaży na rynku brytyjskim znalazł się zapowiadany wcześniej nowy mikrokomputer SPECTRUM PLUS 128 kB. Kosztuje on obecnie ok. 160 funtów brytyjskich.

POWIĘKSZANIE NAPISÓW

Właśnie skończyłeś układać grę, która, jak sądzisz podbije światowe rynki. Brakuje w niej tylko jednego, wspaniałego obrazka, który uprzyjemniałby czas podczas ładowania programu. Dodatkowy problem stwarza fakt, że nie masz pióra świetlnego. Być może jedynym ratunkiem będzie w takim przypadku jeden z czterech poniższych programów.

Wszystkie te programy służą do powiększania dowolnych napisów (również tych zawierających UDG). Tekst przeznaczony do powiększania drukowany jest w dolnej części ekranu, a następnie każdy jego punkt jest sprawdzany funkcją „POINT", która bada stan tego punktu (1 — zamalowany, 0 — nie). Jeżeli „POINT" wynosi 1 w odpowiednim miejscu ekranu powstaje jego powiększona kopia, a następnie cała litera (znak) i napis.

PROGRAM 1 powiększa 8 razy, tworząc napisy z dowolnych znaków (np. ■, =, *, %, itp.)

PROGRAM 2 wykorzystuje właściwości znaków semigraficznych (CHR$ 12 — CHR$ 143). Istnieje mianowicie ścisły związek między „CODE-m" tych znaków a tym, które ćwiartki takiego znaku są zamalowane. „CODE" można wyliczyć na podstawie rysunku

2	1
8	4

dodając 128. Np. „CODE" znaku



wynosi 128 + 2 + 4 = 134. Program ten powiększa 4 razy.

Programy 3 i 4 są bardzo wolne, ale tę ich wadę rekompensują w całości ich zalety. Uzyskiwane napisy mogą mieć dowolną wielkość, proporcje, odstęp między literami, mogą być pochylone, mogą się wznosić lub opadać. W programie 4 jest również wprowadzony element przypadkowości (w pr. 3 może być). Wszystkie te cechy można regulować zmieniając wartości odpowiednich zmiennych. Nie podaję, która zmienna odpowiada za co. Jeżeli zdecydujesz się skorzystać z któregoś z tych programów, kilka minut eksperymentów rozwieje wątpliwości.

I jeszcze jedno: zapomniałbym dodać, że wymienione programy powstały na ZX-SPECTRUM 48K.

Wojtek

```
  10 REM    PROGRAM  1
  20 LET X=1: LET Y=14: LET Z$="$$": LET A$="NEW!"
  30 PRINT INK 0;AT 21,0;A$
  40 FOR N=0 TO LEN A$*8-1
  50 FOR M=0 TO 7
  60 IF POINT (N,M)=1 THEN PRINT AT Y-M,X+N;Z$
  70 NEXT M: NEXT N: STOP
 100 REM    PROGRAM  2
 110 LET X=0: LET Y=12: LET A$="SPECTRUM"
 120 PRINT INK 0;AT 21,0;A$
 130 FOR N=0 TO LEN A$*8-1 STEP 2
 140 FOR M=0 TO 7 STEP 2
 150 PRINT AT Y-M/2,X+N/2;CHR$ (128+POINT (N+1,M+1)+POINT (N,M+1)*2+POINT (N+1,M
)*4+POINT (N,M)*8)
 160 NEXT M: NEXT N
 165 STOP
 200 REM    PROGRAM  3
 210 LET X=0: LET Y=8: LET R1=1: LET R2=1: LET R3=1: LET A$="Sinclair"
 215 LET P1=9: LET P2=4: LET P3=10: LET P4=3: LET P5=0
 220 LET O=-2: LET W1=14: LET W2=0: LET PL=1: LET PP=.1
 225 PRINT AT 21,0; INK 0;a$
 230 FOR N=0 TO LEN A$*8-1
 235 FOR M=0 TO 7
 240 IF N/8=INT (N/8) THEN IF M=0 THEN IF N<>0 THEN LET X=X+O: LET Y=Y+W1
 245 IF M=7 THEN LET Y=Y+W2
 250 IF POINT (N,M)=1 THEN GO SUB 260
 255 NEXT M: NEXT n: STOP
 260 FOR A=0 TO P1: PLOT X+N*P2+RND*R1+M*PL+A*PP,Y+M*P3+A+RND*R2: DRAW P4+RND*R3
,P5: NEXT A: RETURN
 300 REM    PROGRAM  4
 310 LET A$="BAJTEK"
 315 LET X=8: LET Y=8
 320 LET P1=19: LET P2=5: LET P3=20
 325 LET R1=4: LET R2=19
 330 LET O=0: LET W1=0: LET W2=0: LET P=0
 335 PRINT AT 21,0; INK 0;a$
 340 FOR N=0 TO LEN A$*8-1
 345 FOR M=0 TO 7
 350 LET B=2
 355 IF N/8=INT (N/8) THEN IF M=0 THEN IF N<>0 THEN LET X=X+O: LET Y=Y+W1
 360 IF M=7 THEN LET Y=Y+W2
 365 IF POINT (N,M)=1 THEN LET B=30
 370 FOR A=1 TO B: PLOT X+N*P2+RND*R1+M*P,Y+M*P3+RND*R2: NEXT A
 375 NEXT M: NEXT N
 380 STOP
9999 PAPER 0: INK 7: BORDER 0: CLS
```





Fotoreportaż MICHAŁA KUŁAKOWSKIEGO



LEKCJA

Oto jak wyglądają zajęcia w szkole podstawowej w Seget na Węgrzech. Nie jest to — co prawdą — typowa węgierska podstawówka, ale nie jest także jedyną, w której nauczyciele nie przestraszyli się konkurencji mikrokomputerów.

Nauka programowania rozpoczyna się już w trzeciej klasie. Natomiast piątoklasiści na zajęciach z przedmiotu „technika", budują samodzielnie proste roboty współpracujące z komputerem. Z pomocy mikrokomputera bardzo chętnie korzystają nauczyciele innych przedmiotów, także humanistycznych.

Uczniowie polskich szkół mogą jedynie pozazdrościć węgierskim kolegom.

(rp)

NARODZINY MIKROPROCESORA

Kontynuujemy druk wybranych fragmentów wydanej w 1984 roku w Nowym Jorku książki Everetta M. Rogersa i Judith K. Larsen pt. „Gorączka Krzemowej Doliny".

Miejsce akcji: Kalifornia, USA.

Czas akcji: początek lat siedemdziesiątych.

Wraz z wynalezieniem tranzystora w Laboratoriach Bella, oraz układu scalonego — czego dokonali Rob Noyce i Jack Kilby, do najdonioślejszych wynalazków w przemyśle mikroelektronicznym należy mikroprocesor — pochodzące z r. 1971 dzieło Teda Hoffa. Zważywszy na rolę mikroprocesora, może dziwić, że jego wynalazca, dr Marcian E. (Ted) Hoff młodszy nie zyskał szerszego uznania w społeczeństwie.

Gdy spotkaliśmy się z Hoffem, wyczuliśmy, że ten inżynier w średnim wieku woli nie rzucać się w oczy, ponieważ dzięki temu, że nie jest obiektem zainteresowania publiczności może nadal poświęcać się wynajdywaniu nowych rzeczy. O wynalezieniu mikroprocesora opowiada rzeczowo, jak uczeń relacjonujący nauczycielowi, co robił podczas ostatnich wakacji. Pełna skromności postawa Hoffa nie przysłania jednak faktu, że praca nad mikroprocesorem była dlań po prostu wielką frajdą.

Czego się Jaś nauczy...

Hoff wyrastał na wsi w pobliżu Rochester w stanie Nowy Jork, gdzie jego ojciec pracował jako inżynier elektrotechnik. „Zainteresowałem się nauką w bardzo młodym wieku" — mówi Hoff. Podczas studiów w pobliskim Instytucie Politechnicznym Rensselaera (RPI) specjalizował się w elektronice pisząc pracę na temat zjawiska przełączania w tranzystorach.

Hoff nigdy nie wyjeżdżał na zachód dalej, niż wodospad Niagara. Gdy więc wystąpił o stypendium Narodowej Fundacji Nauki na studia II stopnia, jako miejsce, gdzie chciałby studiować, wymienił Uniwersytet Stanforda. Słyszał już o Fredzie Termanie a profesorowie w RPI wyrażali się pochlebnie o kalifornijskiej uczelni. Przybywszy do Palo Alto zapisał się na wykłady z teorii tranzystorów. Wkrótce zaczął uczestniczyć w badaniach nad tranzystorami i uzyskawszy doktorat pracował na uczelni jako młodszy badacz. Po dziesięciu latach uczelni zaraził się jednak bakcylem biznesu: „Chciałem pracować nad jakimś pomysłem, który miałby duże znaczenie gospodarcze. Największą pochwałą jest. gdy za to, co wymyśliłeś, ktoś chce ci zapłacić".

Gdy w r. 1968 powstał Intel, poszukiwano pracowników do wydziału badań. Wykładowcy z uniwersytetu gorąco rekomendowali Hoffa ale Intel zaoferował wpierw pracę komuś z IBM. Następnie zwrócono się do pewnego inżyniera od Fairchilda i do-



piero gdy i ten odmówił, Bob Noyce (jeden z szefów firmy Intel) zatelefonował wreszcie do Hoffa. Tym samym Hoff, który stał się trzecim wyborem Intel na stanowisko w dziale badań, został dwunastym pracownikiem firmy. Pracował w małym laboratorium w jednym z pomieszczeń Intel w Mountain View, dzieląc pokój z pozostałym personelem i ze skrzyniami jakichś dostaw dla Union Carbide, bowiem koncern ten wynajmował drugą połowę budynku.

W owym czasie Intel zajmował się głównie pamięciami półprzewodnikowymi, przyjmował jednak również klientów takich jak Busicom. Był to — upadły już — producent japoński, który chciał, aby w Intel skonstruowano specjalnego rodzaju układ scalony przeznaczony do serii kalkulatorów biurkowych. 20 czerwca 1969 r. grupy inżynierów z Busicom przybyła z Tokio na spotkanie z Hoffem, który tegoż wieczora odleciał na od dawna planowany urlop na Tahiti. Gdy wrócił, Japończycy jeszcze czekali. Gdy przedstawili własny projekt zestawu sześciu bardzo złożonych układów scalonych, Hoff powiedział im, że proponowany przez nich schemat jest — jak na możliwości Intel — zbyt skomplikowany.

Obok biurka Hoffa stał minikomputer PDP-8, którym Hoff posługiwał się w toku swoich badań. Pomyślał o możliwości skonstruowania czegoś, co byłoby jeszcze mniejsze — mikrokomputera. „Patrzyłem na PDP-8, na plany ludzi z Busicom i zastanawiałem się, czemu kalkulator ma być aż tak wiele bardziej skomplikowany. Myślał o umieszczeniu komputera

w jednym układzie scalonym

Wreszcie opracował schemat mikroprocesora, czyli układu scalonego służącego za sterujący komputerem centralny procesor. Konstruując pierwszy w świecie mikroprocesor, Hoff postanowił wszystkie funkcje procesora umieścić na jednym układzie scalonym. Z mikroprocesorem połączył następnie dwie kostki pamięci — jedną do przechowywania danych, zaś drugą do przechowywania kierującego działaniem procesora programu. Miał teraz w ręku elementarny komputer ogólnego przeznaczenia, który mógł pracować nie tylko w złożonym kalkulatorze, lecz w zależności od programu również sterować windą, światłami ulicznymi i wykonywać wiele innych zadań...

Choć dziś może to wydawać się dziwne, ale opracowany przez Hoffa projekt mikroprocesora nie wywarł na japońskich inżynierach wrażenia. Do tego stopnia, że zabrali się do pracy, aby przekonstruować własny układ. Hoff kontynuował jednak swoje prace nad schematami mikroprocesora, mimo że japoński klient Intel nie był nim zainteresowany.

Bob Noyce zdecydowanie go popierał. Już kilka lat wcześniej przewidywał, że do czegoś takiego dojdzie. Na jednej z konferencji pod koniec lat 60-tych, gdy Noyce zapowiedział nadejście komputera mieszczącego się w jednej kostce, jakiś krytykant spośród słuchaczy zareagował na to uwagą: „Nie chciałbym stracić całego komputera przez szparę w podłodze". „Źle mnie pan zrozumiał — odparł Noyce — ponieważ będzie pan miał na biurku jeszcze sto innych, nawet jeśli zgubi pan jeden, nie będzie to mieć żadnego znaczenia".

W owym czasie większość projektantów nie interesowała się możliwościami małych komputerów. Uważali, że prawdziwie ciekawe rzeczy dzieją się przy komputerach uniwersalnych. Hoffowi udało się jednak przekonać niedawno przejętego z Fairchilda Stana Mazora (Stanisława Mazura), aby wraz z nim zajął się mikroprocesorem.

W marcu 1970 r. do Intel przybył z Fairchilda Federico Faggin. On to właśnie opracował architekturę układu i rozrysował obwody. Zdaniem Hoffa, stworzony przez niego projekt był

„piękny i czysty"

Tymczasem w Busicom pracowano nad kalkulatorem, którego sercem był mikroprocesor Intel. W niecały rok później, w styczniu 1971 Hoff i koledzy mieli już działający układ.

Pierwszy mikroprocesor w jednym układzie scalonym otrzymał w Intel nazwę „4004". Przedrostek oznaczał, iż jest to wyrób wykonany na zamówienie i przeznaczony dla firmy Busicom. Końcówka natomiast sygnalizowała, że jest to czwarty układ zaprojektowany na zamówienie przez Intel.

Prezes Intel, Gordon Moore opisał możliwości mikroprocesora w następujących słowach: „Możemy teraz wytwarzać jeden rodzaj układu i sprzedawać go do paru tysięcy różnych zastosowań". To właśnie jest piękno mikroprocesora — może on wchodzić w skład dowolnego wyrobu elektronicznego, który chce się wyposażyć w zminiaturyzowaną moc obliczeniową. Owa elastyczność zastosowań pociągała za sobą niezmiernie dalekie konsekwencje natury handlowej. Ted Hoff wynalazł sposób, dzięki któremu Intel miał szybko zrobić majątek.

W październiku 1969 r. Victor Poor z firmy Computer Terminals Corporation zwrócił się do Intel z zamówieniem na wyspecjalizowany układ scalony do budowanej przezeń końcówki komputerowej. Hoff i Mazor skonstruowali wspólnie mikroprocesor o większej pojemności niż 4004. Nazwali go Intel 8008. Gdy Mazor pokazał go Poorowi, ten był zdumiony, bowiem nie wierzył, że komputer można umieścić na układzie scalonym. Produkcję mikroprocesora 8008 podjęto w roku 1972.

Wydział handlowy Intel nie był przekonany, że mikroprocesory utrzymają się na rynku i że warte są kosztów i wysiłku wkładanego w ich produkcję. Mimo, że Intel sprzedawał 4004 Busicomowi od stycznia 1971 r., nie oznajmiono o jego produkcji publicznie, skutkiem czego na mikroprocesor ten nie było popytu. W miarę, jak mijał rok 1971, Hoff, Mazor i pracujący z nimi niewielki zespół entuzjastów mikroprocesora chodzili do ludzi od marketingu i nalegali, aby o istnieniu 4004 oznajmiono publicznie.

W listopadzie 1971 r. Intel ostatecznie zdecydował się oznajmić publicznie o 4004. Zakupiono w tym celu ogłoszenie w branżowym piśmie ELECTRONIC NEWS. Ogłoszenie mówiło nie o po prostu jakimś nowym wyrobie, lecz o „nowej erze elektroniki układów scalonych... mikroprogramowanym komputerze na jednej kostce". Firmom półprzewodnikowym zdarzało się już w przeszłości czynić wygórowane obietnice, reakcja na 4004 była więc pełna ostrożności. W typowy sposób zachował się pewien klient, który jesienią 1971 r., podczas wystawy komputerów w Las Vegas podziwiał przy stoisku Intel... nerwy ludzi twierdzących, że mają komputer na jednym scalaku. Dopiero, gdy Mazor pokazał mu specyfikację techniczną, musiał z niechęcią przyznać, że

to naprawdę komputer

W połowie roku 1972 Ted Hoff i Stan Mazor ruszyli na trzy tygodnie w drogę organizując seminaria i spotkania z konstruktorami układów scalonych z różnych firm. Entuzjaści mikroprocesora z Intel zetknęli się z wieloma wątpliwościami. Jednym z najczęstszych pytań było: „No dobrze, a jak to się naprawia". Ludzie nie mogli pojąć, że komputer można po użyciu wyrzucić tak jak przepaloną żarówkę. Hoff przypomina, że na owym wczesnym etapie rozwoju mikroprocesora wielu klientów chciało za cenę mikroprocesora mieć moc obliczeniową minikomputera. „Tym czasem nie mogliśmy jeszcze dorównać minikomputerom (Jeszcze nie)".

Hoff, Mazor i Faggin nadal pracowali nad ulepszeniem mikroprocesora. Efektem tego w sierpniu 1973 stał się mikroprocesor Intel 8080. Zaczęło się od drobnej przeróbki układu 8008, ruszenie schematu wymagało jednak i tak opracowania nowych masek, więc Hoff i Faggin postanowili przebudować układ całkowicie. „W tym właśnie czasie zaczęła wchodzić technologia N-MOS, wykorzystaliśmy więc ją w 8080". MOS, to skrót „metal-oxide on silicon" — tlenek metalu na krzemie", odmiana technologii mikroelektronicznej stosowanej w produkcji układów scalonych bardzo dużej integracji.

„Pod względem wielkości zbytu 8080 jest jednym z największych bestsellerów wszechczasów. Poleciało jak lawina" — mówi Hoff. W miarę jak doświadczenie zdobywane przez firmę w toku produkcji mikroprocesora pomagało przechodzić w dół „krzywej uczenia się", cena 8080, która początkowo wynosiła 360 dol., zaczęła spadać. „Dziś" — mówi Hoff — „przy zakupie hurtowym 8080 kosztuje około 2,5 dol. za sztukę, a wydziałem nawet pojedyncze egzemplarze sprzedawane w sklepach dla hobbystów po 2,95 dol. „Żartując Hoff wspomina o „prawie Gelbacha" — od nazwiska byłego szefa Wydziału Handlu, obecnie zaś wiceprezesa Intel. Prawo to głosi, iż „każdy scalak będzie kiedyś kosztował 5 dolarów — z wyjątkiem tych, które będą tańsze".

SPRZĘŻENIE ZWROTNE

PISALI O BAJTKU

TRYBUNA LUDU:

Ukazał się w nowej szacie graficznej „Bajtek" — comiesięczny dodatek do „Sztandaru Młodych".

Ambicją zespołu redakcyjnego jest — jak czytamy — zwalczanie analfabetyzmu mikrokomputerowego w Polsce. Miłośnicy informatyki, zarówno ci początkujący jak i bardziej zaawansowani znajdą w „Bajtku" coś dla siebie. Jak przygotować sprzęt do pracy, jakie są możliwości trójwymiarowej grafiki komputerowej, zastosowania mikrokomputerów w gospodarce, informatyka na świecie — to tylko niektóre tematy.

W „Bajtku" jest także coś dla przedszkolaków: Kubuś Literka czyli maluchy bawią się w pisanie gry komputerowej.

Miesięcznik jest strzałem w dziesiątkę. Spontaniczny, społeczny ruch wokół informatyki, dziesiątki klubów mikrokomputerowych, które powstają w całej Polsce od dawna czekały na „Bajtka". Dostęp do fachowo opracowanej informacji, dane o podstawowych językach programowania, poznanie możliwości komputerów, to jedna z funkcji „Bajtka" w szerzeniu kultury technicznej, nie tylko wśród młodzieży.

86.03.14

RAZEM:

„Bajtek" wrócił! I to jak — w kolorze, większym formacie i objętości, a także w nakładzie 200-tysięcznym! Powrót to więc triumfalny, ale nie zapominajmy, że bez „starego" „Bajtka" i jego sukcesu rynkowego nie byłoby obecnego wydawnictwa, przygotowanego przez redakcję „SM". Wydaje nam się, że **każdy miłośnik mikrokomputerów w Polsce** (ich liczba gwałtownie wzrasta) **powinien zostać czytelnikiem „Bajtka".** Chwaląc konkurencję przypominamy, że i u nas rubryka „MIKRO" oferuje moc atrakcji.

86.03.16

KURIER POLSKI:

— Bez komputera nie ma jutra — usłyszał jeden z moich znajomych od swej nastoletniej latorośli, która po wypowiedzeniu tego historycznego stwierdzenia zażądała gotówki, by w kioskach „Ruch" rozpocząć polowanie na miesięczny dodatek do „Sztandaru Młodych" — „Bajtek".

Właśnie ukazał się pierwszy tegoroczny, kolorowy numer, w którym każdy nastolatek zauroczony, a który z nich nie jest, mikrokomputerami znajdzie coś dla siebie. M.in. dowie się, iż programować w języku logiki może każdy, jak robią to inni, o gorączce Krzemowej Doliny.

Nam kilkudziesięciolatkom, pozostaje tylko zazdrościć. My nie mieliśmy „Bajtka".

86.03.19



NIE BÓJ SIĘ MNIE!



Serwus! Jesteśmy już starymi przyjaciółmi. Wiesz o mnie sporo, potrafisz połączyć mnie z magnetofonem i telewizorem, uruchomić, a ostatnio próbowałeś nawet nagrać program z kasety.

Magnetofon i kaseta z taśmą to najprostszy i najtańszy rodzaj mojej pamięci zewnętrznej. Przyznam Ci się jednak, że nie przepadam za tym rozwiązaniem. Najchętniej pracuję ze stacją dyskietek (elastycznych krążków, które stanowią skrzyżowanie płyty gramofonowej z taśmą magnetofonową). Przyczyn jest kilka. Ot, choćby ta, że wczytując program z dyskietki mylę się niezwykle rzadko, podczas gdy z magnetofonem zdarza mi się to o wiele częściej. Przyczyną jest między innymi niewielka trwałość kaset.

Drugą przewagą dyskietki jest znacznie większa szybkość wczytywania programów, choć jeśli chodzi o mego kuzyna — Commodore 64 różnica ta bywa niewielka. I jest jeszcze jedna — chyba najważniejsza — przyczyna mojej sympatii dla dyskietki... ale najpierw zastanówmy się wspólnie w jaki sposób mogę korzystać z zapisu na kasecie magnetofonowej.

Jeden z tych sposobów już znasz, jest to po prostu odczytywanie programu. Załóżmy jednak, że podczas jego wykonywania będę potrzebował jeszcze jakichś dodatkowych informacji, jak np. tabele liczb, listy słów itd. Możesz zapisać je sobie osobno i na moją prośbę ponownie uruchomić magnetofon, tym razem z taśmą zawierającą dane do obliczeń. W ten sposób konstruowane są zwykle wszelkie katalogi czy bazy danych. Nie byłoby żadnego problemu, gdyby nie fakt, że taki na przykład katalog książek, nie zmieści się na raz w mojej pamięci. Jej pojemność jest ściśle określona i nie ma rady — choćbyś nawet próbował dopchnąć kolanem — nic więcej nie wciśniesz. W tym przypadku programiści radzą sobie tak: każą mi najpierw wczytać część danych, wyszukać informacje, które są mi potrzebne, zapamiętać je, a całą resztę zapomnieć. Teraz dopiero mogę wczytywać następną część.

Wszystko jest w porządku, do chwili, gdy będzie mi potrzebna nowa informacja „z tyłu", to znaczy z fragmentu, który był już wczytywany. Tak się dzieje na przykład, jeśli zadasz mi pytanie o kolejną książkę z Twojego katalogu. W tym momencie musisz znowu cofnąć taśmę a zabawa zaczyna się od nowa. Zwróć uwagę, że czasem dla znalezienia innej informacji muszę „przesłuchać" cały zapis, a trwa to od kilku minut do... To już zależy od wielkości Twojej biblioteki.

Jeśli jednak podłączysz mnie do stacji dyskietek, sytuacja wygląda zupełnie inaczej. Każdą informację mogę znaleźć w czasie wystarczającym na jeden pełny obrót elastycznego krążka i z każdej mogę korzystać wielokrotnie. Oczywiście pod warunkiem, że wpierw zapamiętam, w którym miejscu na dyskietce zapisane są jakie informacje. Ale to już naprawdę drobiazg.

Istnieją urządzenia pozwalające na jednoczesne korzystanie z informacji zapisanych na dwóch i więcej dyskietkach. Dotyczy to jednak raczej moich bardziej poważnych kolegów — komputerów osobistych. Ja jestem przecież tylko zwykłym domowym komputerkiem. Nie mogę również marzyć o takich luksusach jak twarde dyski, pozwalające zapisać naprawdę niewiarygodne ilości informacji, czy laserowy zapis i odczyt danych.

Właśnie. Jak się nie ma co się lubi, to się lubi co się ma. Miło było dowiedzieć się jak dobrze jest korzystać ze stacji dyskietek, lecz powróćmy na ziemię, to znaczy do naszego magnetofonu i kasety z taśmą. Widzę, że moim opowiadaniem zepsułem Ci nieco humor. Nie przejmuj się jednak. Możesz — jeśli tylko ruszysz głową — ominąć wiele niedogodności, o których Ci mówiłem. Przede wszystkim musisz starannie obchodzić się z kasetami. Co to znaczy? Musisz po prostu dbać o nie tak jak o te, których używasz do nagrywania muzyki, tylko... o wiele bardziej.

Na czym polega różnica? To proste. Jeśli np. taśma z najnowszym nagraniem Michaela Jacksona wskutek Twojej nieuwagi lub niesprawności magnetofonu — zagnie się w jakimś miejscu, to podczas odtwarzania muzyki usłyszysz trzask lub zniekształcenie i dalej wszystko będzie w najlepszym porządku. Jeśli natomiast taki wypadek zdarzy mi się podczas odczytywania programu, to wówczas najczęściej, nie da się go uruchomić.

Z pewnością świetnie wiesz jak powinno się używać kaset i w jakich warunkach je przechowywać. Na wszelki wypadek przypomnę Ci jednak kilka podstawowych zasad. Zawsze trzymaj je w szczelnych, oryginalnych pudełkach. Po zakończeniu pracy nie zostawiaj kasety w magnetofonie, gdyż i tam dociera wszędobylski kurz. Korzystaj tylko ze sprawnego magnetofonu, pamiętaj, że nawet mała niesprawność mechaniczna może spowodować niepowetowaną stratę — zniszczenie jakiegoś cennego programu.

Radzę Ci używać zawsze tego samego magnetofonu, zarówno do nagrywania jak i do odtwarzania programów — niewielkie różnice w ustawieniu głowicy często przysparzają sporo kłopotów. Co do wyboru konkretnego magnetofonu to nie ma na to ustalonych reguł. Praktyka wykazuje, że najlepiej współpracuje mi się z prostymi magnetofonami np. „kapral". Dla Twojej wygody przydaje się też licznik obrotów. Łatwo wówczas znaleźć poszukiwany program.

Moi kuzyni, Commodore i Atari są bardziej wybredni — nie chcą współpracować z magnetofonami innymi niż ich własne. A inny mój krewniak, Schneider 464 ma nawet magnetofon wmontowany w obudowę. Przy okazji kilka słów o zaletach i wadach tego rozwiązania. Pierwsza wada jest oczywista, taki magnetofon trzeba po prostu kupić, podczas, gdy chyba każdy ma w domu jakiś zwykły „kaseciak", który można by wykorzystać. Zaletą jest natomiast to, że Atari czy Commodore mogą same włączać i wyłączać magnetofon w razie potrzeby. Niestety, prawie we wszystkich przypadkach sterowanie to polega na włączaniu i wyłączaniu silnika. Gumowe kółko dociskające taśmę do rolki napędowej nie podnosi się, co powoduje zagniecenie — zawsze w tym samym miejscu, na końcu programu. Jeśli z drugiej strony taśmy nagrany jest w tym miejscu program, może on po pewnym czasie przestać się wczytywać. A więc dwie praktyczne rady: Programy nagrywaj tylko z jednej strony kasety, a gdy nie korzystasz z magnetofonu, wyłączaj go.

Tego, że kasety nie należy zbytnio nagrzewać — a więc kłaść jej na kaloryferze czy też wystawiać na słoneczko — oraz kłaść na głośnikach (w pobliżu magnesów i cewek) nie muszę Ci chyba przypominać.

Dotychczas mówiłem prawie wyłącznie o moich wymaganiach. Porozmawiajmy sobie teraz o Twojej wygodzie. Gdy masz kilka czy kilkanaście programów, znalezienie któregoś z nich nie nastręcza większych kłopotów. Nawet jednak nie obejrzysz się, gdy nazbiera Ci się ich tak wiele, że nie połapiesz się w tym bałaganie. Wiele razy będziesz sobie obiecywał, że zrobisz z tym porządek i nigdy nie znajdziesz na to czasu.

Chyba, że... od samego początku zaprowadzisz idealny porządek. Nie żałuj pieniędzy na kasety — nagrywaj osobno programy użytkowe, osobno gry, osobno języki programowania i osobno programy własne. Pozostaw też jedną czy dwie kasety na „brudnopisy" przeznaczone na programy, które właśnie piszesz. Warto by także pomyśleć o utworzeniu banku rezerwowych nagrań najbardziej cennych programów. (Takie — rzadko używane — kasety należy „dla zdrowia" raz w miesiącu przewinąć na drugą stronę).

I jeszcze jedno. Nigdy nie nagrywaj programów na stare, zużyte kasety. Oszczędność to tylko pozorna. Więcej zdrowia stracisz próbując po jakimś czasie odtworzyć program z takiej kasety.

Cierpliwości i staranności życzy Ci

Twój Komputer



ATARI W KRAKOWIE

Historia Atari w naszym mieście rozpoczęła się praktycznie w lutym 1985 r. kiedy to udało mi się namówić kilku przyjaciół na zakup mało wtedy popularnego komputera.

Klub nasz jest sekcją krakowskiego Klubu Użytkowników Mikroprocesorów, który powstał w kilku miastach w ramach Naczelnej Organizacji Technicznej a ściślej SEP-u. Nie będę przedstawiał szczegółowego kalendarium naszego rozwoju, lecz ograniczę się do kilku znaczących informacji:

Posiadamy kontakty z 58 osobami i klubami zagranicznymi, głównie z USA, ale także z Kostaryki, Australii i Malty.

Prenumerujemy następujące pisma: Compute, Antic, Analog, Atarien, Atari User, Monitor, Atari Connection, A-Z of Personal Computers. Zgromadziliśmy biblioteczkę składającą się z 67 pozycji, w tym najbardziej wartościowe prace dotyczące mikroprocesora 6502 oraz kompletną dokumentację serwisową.

Współpracujemy z Pewexem w Krakowie i P.Z. Karen z Warszawy. Zorganizowaliśmy serwis, który prowadzi usługi gwarancyjne i naprawy pogwarancyjne na terenie Polski południowej.

Klub nasz został wyposażony przez firmę Karen w sprzęt (800XL, magnetofon, telewizor Venus), a w najbliższych dniach mamy otrzymać 2 stacje dysków i drukarkę.

Napisaliśmy i wydaliśmy (dla potrzeb klubowych) książkę „ATARI BASIC". W chwili obecnej staramy się wydać jej rozszerzoną wersję, która zawierać będzie m.in. dokładny opis rozkazów procesora 6502 wraz z komentarzami i może służyć do nauki programowania także dla użytkowników Commodore i Apple.

Opracowaliśmy kilka ciekawych urządzeń peryferyjnych do Atari (m.in. interface do zwykłego magnetofonu, pióro świetlne, klawiaturę numeryczną i digitizer).

Zebraliśmy ponad 2000 programów, w tym wiele użytkowych jak VisiCalc, Word Processor, dBase. Dysponujemy następującymi językami programowania: kilka wersji języka BASIC (także kompilator), LOGO, Forth, Pascal, Pilot, Action, C, Assembler, Macroassembler/Editor, Mac-65. Posiadamy również wiele programów narzędziowych.

W tej chwili pracujemy nad następującymi problemami: Podręcznik do nowej wersji języka BASIC, którą usilnie propagujemy. BASIC XL jest nowoczesnym interpreterem, ocenianym wyżej niż BASIC Microsoft. Jest on w jedną stronę kompatybilny z Atari Basic, a zarazem kilkakrotnie od niego szybszy. Posiada 46 instrukcji więcej m.in. ENDWHILE, ENDIF, ELSE, WHILE, TRACE, NUM, RENUM, binary GET i PUT oraz double POKE i PEEK. Przy tym wszystkim, dzięki świetnemu pomysłowi, pozostawia dla użytkownika tyle samo wolnego RAM-u co BASIC resident, mimo że sam zajmuje 16 kB.

Zamierzamy również opracować suplement do podręcznika M. Iglewskiego „Pascal", który zawierałby różnice w stosunku do stosowanego przez nas Kyan Pascala oraz programy przykładowe w tym języku.

W związku z niebywałym wzrostem popularności Atari rozpoczęliśmy już wstępne prace nad opracowaniem zestawu programów z zakresu statystyki i ekonometrii oraz rachunkowości.

Bardzo interesujące, choć czasochłonne, są próby wykorzystania Atari jako sterownika do urządzeń pomiarowych i maszyn.

Będziemy pisać program na komputery serii XL i XE do przetwarzania tekstów z 80-cio kolumnowym generatorem polskich znaków oraz z printer drive'm do drukarek D100 i DZM.

Dążymy do tego, by w przyszłości pracować na prawdziwie profesjonalnym sprzęcie serii ST.

Tak przedstawiają się nasze główne osiągnięcia i plany. Ich realizacja jest kwestią czasu, włożonej pracy i odrobiny szczęścia. Istnieje jednak problem, którego nie jesteśmy w stanie rozwiązać sami. Z jednej strony cieszymy się, że wzrasta liczba użytkowników Atari w Polsce, z drugiej zaś zdajemy sobie sprawę, że wymaga to szerokiej akcji popularyzatorskiej i nieomal ciągłego strumienia informacji i oprogramowania. Po opublikowaniu artykułu w Przeglądzie Technicznym dostaliśmy ponad 200 listów z najróżniejszymi pytaniami. Stopniowo postaramy się odpowiedzieć na wszystkie listy, nie jesteśmy jednak instytucją i nie mamy tak dużych możliwości biurowych ani czasowych by zrobić to szybko i wyczerpująco. Trzon grupy stanowią studenci i młodzi pracownicy naukowi AGH.

Adres do korespondencji:
Klub Użytkowników Atari
Skr. pocztowa 375
30-950 Kraków 61

Wiesław Migut

ANKIETA

INDYWIDUALNY BANK DANYCH

W poprzednim numerze Bajtka opublikowaliśmy ankietę skierowaną do klubów mikrokomputerowych. Dzisiaj zwracamy się do indywidualnych użytkowników mikrokomputerów. Tworzymy bank danych mający służyć wszystkim naszym czytelnikom.

Najbardziej interesują nas rzadziej spotykane w naszym kraju mikrokomputery. Ich właściciele zasypują redakcję prośbami o kontakt z innymi użytkownikami. Nie oznacza to jednak, że mniej zależy nam na informacjach o posiadanych przez Was Commodore czy Spectrum.

Jeśli nie będziesz mógł przysłać ankiety wyciętej z BAJTKA, napisz do nas po prostu na zwykłej kartce.

Imię i nazwisko: ....

Adres: ....

Telefon: ....

Zawód, wiek: ....

Zainteresowania: ....

Posiadany mikrokomputer: ....

Urządzenia peryferyjne: ....

Oprogramowanie własne: ....

firmowe: ....

ANKIETA

Literatura:

Czasopisma:

Co masz do zaproponowania innym użytkownikom?

Czego spodziewasz się po kontakcie z innymi użytkownikami?

Co chciałbyś przeczytać w BAJTKU?

Uzupełnienie informacji:

Wypełnij! Wytnij! Wyślij! Nasz adres: „Bajtek", 00-687 Warszawa, ul. Wspólna 61

SPRZĘŻENIE ZWROTNE

Drogi Bajtku!

Ciekawi jesteśmy zdania czytelników na temat obecnej objętości, formy, a przede wszystkim — jakości naszego czasopisma. Prosimy o listy na ten temat, uwagi, którymi być może pragnęliby się z nami podzielić.

Jednocześnie dziękujemy za dotychczas nadesłane listy.

Próbując skuteczność zabezpieczenia programów przed skopiowaniem (Bajtek nr 1), zabezpieczyłem program gry „PUNCHY". Niedawno kolega zaproponował mi wymianę na grę „MEXICO". Nie mogę się z nim wymienić. Jedyna możliwość to kupić od niego program, ale przecież mógłbym się wymienić. Chciałbym wiedzieć, jak cofnąć blokadę. Proszę, pomóżcie!

Radosław Fryczkowski
ul. Sandomierska 11/96
26-611 Radom

Niestety, żal naszego czytelnika jest przysłowiową musztardą po obiedzie. Nie możemy podać sposobu likwidacji blokady; artykuł „Ochrona programów" z nru 1 straciłaby przez to swą wartość.

Będziemy też na łamach Bajtka propagowali raczej zabezpieczenie programów, niż łamanie zabezpieczeń — w Polsce i tak rynek oprogramowania opiera się niemal wyłącznie na produkcji „pirackich" kopii, co nie powinno trwać bez końca.

Jak każde zabezpieczenie, również to podane w nrze 1 da się zlikwidować. Sądzę, że ta okoliczność stanowić może dobry pretekst do poznania kodu maszynowego.

Do napisania listu zmusiła mnie lektura artykułu „Coś dla piratów" z nru 3. Chodzi o przedstawiony tam program kopiujący szumnie nazwany najkrótszym. Oto moja propozycja programu, który daje identyczne rezultaty jak ów „najkrótszy program kopiujący" — radzę spróbować!

```
F3  DI      ; wyłącz przerwania
76  HALT    ; stop
```

Efekt działania obu programów jest identyczny, czyli żaden.

Błąd jest prozaiczny, i polega na tym, że w słowie przeczytanym przez procesor z buforu wejściowego o adresie 254, tylko jeden bit (a nie cały bajt) niesie informację o stanie wejścia EAR!

Analogicznie w słowie wysłanym na wyjście tylko jeden bit ustawia poziom logiczny na wyjściu MIC. Jest tylko jeden szkopuł — to nie jest ten sam bit!

Oto bajt przeczytany przez instrukcję IN:

7	6	5	4	3	2	1	0
-	EAR	-	d4	d3	d2	d1	d0

gdzie: d0–d4 — stany odpowiednich półrzędów klawiatury
– — bity niewykorzystane
EAR — stan wejścia EAR

A to bajt wysłany instrukcją OUT:

7	6	5	4	3	2	1	0
-	-	-	BUZZ	MIC	G	R	B

tu z kolei GRB — kolor BORDER
MIC — informacja wysyłana na wyjście MIC
BUZZ — informacja wysyłana na głośnik
– — bity niewykorzystane

Jak widać bit przeczytany z wejścia znajduje się na pozycji 6, więc zostaje wysłany „w pustkę" instrukcją na OUT.

Więc dlaczego w ten sposób można kopiować programy, mimo że program nie działa? Otóż wynika to z faktu, że między wejściem EAR a wyjściem MIC oraz głośnikiem znajduje się pętla RC. Dla częstotliwości odczytywanych z magnetofonu ma ona charakter różniczkujący i stąd pozorna „poprawa jakości nagrania", polegająca na podbiciu wyższych częstotliwości sygnału. Ale do tego sposobu kopiowania wcale nie jest potrzebna ingerencja programowa komputera. Więc można bez obaw „zawiesić" komputer, otrzymując identyczne efekty, gdyż sygnał z wejścia EAR swobodnie dostanie się na wyjście MIC. Jednak opisany sposób niewiele się różni od bezpośredniego połączenia ze sobą dwóch magnetofonów.

Autora programu chciałbym pocieszyć, że pomyłki w programach popełniają również zawodowi programiści — i tak np. ZX Spectrum zawiera w swoim ROM-ie aż 13 błędów! (...)

Leszek Zdonek
ul. Sportowa 6/1
44-100 Gliwice

Wynajdywanie błędów w programach jest zajęciem pasjonującym, zwłaszcza, gdy poparte jest szczegółową analizą ich działania, tak jak w tym przypadku. Dziękujemy — liczymy na podobne listy w przyszłości!

Warto przypomnieć, że opisany w Bajtku nr 3/85 program był przedrukiem z miesięcznika „Your Computer" — co było zresztą uwidocznione w podpisie — a więc nie tylko my daliśmy się na to nabrać.

Zgodnie z wezwaniem Bajtka zgłaszamy swoje istnienie. Nasz klub, działający przy L.O. w Witnicy nosi nazwę „Sezam Mądrości".

Mimo krótkiego okresu działalności (półtora miesiąca) zebraliśmy już fundusze na pierwszy komputer. Prawdopodobnie zakupimy go w przeciągu najbliższych tygodni. Jednak na tym nie poprzestaniemy mamy większe aspiracje. Liczymy na pomoc oświaty i zakładu opiekuńczego.

Chcielibyśmy, przy Waszej pomocy, nawiązać kontakt z innymi klubami mikrokomputerowymi, które, podobnie jak my postanowiły same zapracować na sprzęt.

Członkowie klubu
„Sezam Mądrości"
Liceum Ogólnokształcące
w Witnicy, woj. gorzowskie

Ten list nie wymaga komentarza. Bajtek życzy powodzenia!

Marcin Waligórski



LICZYDEŁKO

8·8=64
500+100=600
16:4=4

Cześć Maluchy!

Dzisiaj coś dla leniuchów. Tych, którym nie chce się liczyć. Komputer Wam pomoże. Ale nie ma nic za darmo. Trzeba nauczyć się korzystać jeszcze z kilku instrukcji.

Nasz program będzie dodawał, odejmował, mnożył i dzielił. A więc na początku komputer musi nas poinformować jaki mamy wybór. Napiszmy więc:

```
10 CLS
20 PRINT "                LICZYDELKO"
30 PRINT
40 PRINT "  WYBIERAJ:
50 PRINT
60 PRINT "DODAWANIE    +"
70 PRINT "ODEJMOWANIE  -"
80 PRINT "MNOZENIE     *"
90 PRINT "DZIELENIE    :"
100 PRINT
```

Instrukcja **CLS** oznacza — jak zapewne pamiętacie — czyszczenie całego ekranu. Następnie stosujemy tylko instrukcje PRINT. Przypominam, że rozkaz:

numer linii **PRINT**

oznacza po prostu pozostawienie wolnego wiersza.

Jak jednak powiedzieć komputerowi, że chcemy np. dodawać a nie mnożyć. Nie ma rady! Musimy poznać jeszcze jeden rozkaz — **INPUT**.

numer linii **INPUT** zmienna

oznacza, że komputer ma zatrzymać wykonywanie programu i czekać na to, co wprowadzimy poprzez klawiaturę. Jeśli teraz coś napiszemy i wciśniemy **ENTER**, wówczas nasz komputer zapamięta to „coś" pod nazwą zmiennej.

Nie bardzo rozumiecie co to jest zmienna? Już wyjaśniam. Wyobraźcie sobie, że na stole stoi kilka pudełek, do których przyczepiono karteczki z literami A, B, C,... Do każdego pudełka coś wkładamy. Np. do A — piłkę, do B — klocek, do C — pluszowego misia. Od tej chwili jest nam wszystko jedno, czy prosimy kolegę by dał nam misia, czy też zabawkę z pudełka C. Oczywiście zawartość pudełek może się zmieniać. Np. zamiast misia możemy umieścić w pudełku C lalkę.

Takimi właśnie pudełkami są zmienne. Możemy określać je prościej niż przy użyciu rozkazu **INPUT**. Wystarczy napisać:

X=5

lub

LET X=5

co oznacza, że od tej chwili X będzie oznaczało liczbę 5. Możesz to sprawdzić pisząc:

PRINT X

otrzymasz wynik:

5

A teraz coś trudniejszego:

X=X+2

To nie błąd! Taki rozkaz oznacza, że nowa wartość zmiennej X będzie się równać starej wartości powiększonej o dwa.

Nazwy zmiennych wymyślamy sami. Ich największa długość zależy od rodzaju mikrokomputera. Zwykle nie jest mniejsza niż kilka znaków. Nazwa może się składać z liter i cyfr, ale musi rozpoczynać się od litery. Oczywiście musi to być jedno słowo.

Oto kilka dobrych nazw:

x, X, X1, DOM1,

a oto kilka złych:

1, 1X, DOM 1, RUNDA,

Ostatnia nazwa jest również nieprawidłowa, ponieważ zawiera słowo kluczowe (rozkaz) **RUN**.

Dotychczas mówiliśmy o zmiennych pod którymi ukrywały się liczby. Zmienną możemy stosować także do przechowywania tekstu (łańcuchów) czyli ciągów liter, liczb i innych znaków. Pamiętaj jednak, że na końcu nazwy takiej zmiennej umieszczamy znak $, np.:

a$, ZDANIE$, SLOWO1$, ZNAK$

No więc nareszcie możemy dopisać do naszego programu:

```
110 INPUT ZNAK$
120 PRINT
130 INPUT "PIERWSZA LICZBA   ";PL
140 INPUT "DRUGA LICZBA      ";DL
```

Zwróćcie uwagę na linie 130 i 140. Komputer nie tylko czeka na naszą odpowiedź, ale przypomina nam co mamy wpisać. Warto zapamiętać ten sposób użycia **INPUT**. Nie zapomnijcie o średniku.

Teraz obliczymy wynik naszego działania. Użyjemy do tego instrukcji:

IF warunek **THEN** polecenie

Oznacza ona, że jeśli „warunek" jest spełniony wówczas komputer ma wykonać „polecenie". Napiszmy:

```
150 IF ZNAK$="+" THEN LET WYNIK=PL+DL
160 IF ZNAK$="-" THEN LET WYNIK=PL-DL
170 IF ZNAK$="*" THEN LET WYNIK=PL*DL
180 IF ZNAK$=":" THEN LET WYNIK=PL/DL
```

Widzicie więc, że zmienna WYNIK będzie przyjmowała różne wartości w zależności od tego jakie działanie wybraliśmy.

Teraz należy tylko ładnie zapisać wynik:

```
190 PRINT
200 PRINT PL;ZNAK$;DL;"=";WYNIK
210 PRINT
```

W linii 200, po rozkazie PRINT znalazły się zarówno zmienne liczbowe (PL, DL, WYNIK) jak i tekstowe (ZNAK$) a także jednoelementowy łańcuch („="). Oddzielamy je średnikiem, co powoduje, że będą drukowane jedne za drugimi. Jeśli użyjecie przecinka komputer pozostawi odstępy o szerokości kilku znaków (zależnie od rodzaju mikrokomputera).

Dopiszmy na koniec:

```
220 INPUT "WCISNIJ ENTER";A$
230 GOTO 10
```

Zastanów się jaką funkcję spełnia linia 200.

Program gotowy. Możesz sprawdzić, czy Twój komputer zna tabliczkę mnożenia. Ale sam także nie zapominaj ile jest 7*8.

Romek

```
10 CLS
20 PRINT "                LICZYDELKO"
30 PRINT
40 PRINT "  WYBIERAJ:"
50 PRINT
60 PRINT "DODAWANIE    +"
70 PRINT "ODEJMOWANIE  -"
80 PRINT "MNOZENIE     *"
90 PRINT "DZIELENIE    :"
100 PRINT
110 INPUT ZNAK$
120 PRINT
130 INPUT "PIERWSZA LICZBA   ";PL
140 INPUT "DRUGA LICZBA      ";DL
150 IF ZNAK$="+" THEN LET WYNIK=PL+DL
160 IF ZNAK$="-" THEN LET WYNIK=PL-DL
170 IF ZNAK$="*" THEN LET WYNIK=PL*DL
180 IF ZNAK$=":" THEN LET WYNIK=PL/DL
190 PRINT
200 PRINT PL;ZNAK$;DL;"=";WYNIK
210 PRINT
220 INPUT "WCISNIJ ENTER";A$
230 GOTO 10

READY.
```

32

HORYZONTY KOSMOSU

dze, obrazowo mówiąc, postępu „kosmicznego". Tym właśnie były pierwsze spotkania na orbicie i wystrzelenie pierwszej kosmicznej stacji orbitalnej, lądowanie stacji automatycznych na Księżycu, Wenus i Marsie, realizacja wieloletniego programu „Interkosmos" oraz „Sojuz-Apollo" program „Wega" długotrwałe loty na pokładzie stacji orbitalnych itd. Wreszcie różnorodne prace w otwartym Kosmosie.

Dziś trudno wyobrazić sobie ludzkość bez lotów w Kosmos. I to nie dlatego, że się do nich przyzwyczajono, a dlatego, że wiele naszych ziemskich spraw ściśle związanych jest z działalnością kosmiczną. Np. dzięki temu, że w ciągu 90 minut można okrążyć Ziemię na wysokości 300—350 km, kartograficzne rzuty powierzchni ziemskiej można wykonywać znacznie szybciej niż jakieś 20 lat temu. Zaledwie kilka minut pracy aparatu fotograficznego MKF-6 na orbicie okołoziemskiej zastępuje 2-letni proces tradycyjnego fotografowania Ziemi z pokładu samolotu.

Kosmonauci doskonale widzą w morzu skupiska planktonu — na tle szarobłękitnej wody wyraźnie odcinają się zielonkawe rozpływające się plamy. W pobliżu brzegów plankton czasami tworzy jasnozielone smugi, wokół wysp bardzo przypominające szmaragdowe naszyjniki. A tam, gdzie jest plankton tam są i ryby — to od dawna już wiedzą rybacy.

Kosmonautyka ma także inne znaczenie dla gospodarki narodowej. Chociażby sama nieważkość stwarzająca dla organizmu ludzkiego wiele problemów, pozwala na bardziej efektywne niż na Ziemi wykonywanie wielu procesów technologicznych. Przecież istnienie lub brak siły ciężkości wpływa na warunki powstawania kryształów, na rozkład składników uzyskiwanych materiałów, na stopień czystości i rozłożenia różnych substancji.

Na radzieckich stacjach orbitalnych opanowano wytwarzanie kryształów półprzewodnikowych o niepowtarzalnych właściwościach mających duże znaczenie dla mikroelektryki. Jak wynika z obliczeń ekonomicznych, wygląda na to, że produkcja kosmicznych półprzewodników będzie rentowna.

Duże perspektywy mają też otrzymywane w stanie nieważkości superczyste, biologicznie aktywne substancje. Mogą one być wykorzystywane do produkcji cennych środków medycznych, a także w przemyśle mikrobiologicznym.

Zaledwie pół roku temu prace z urządzeniem „Tawrija" mnie samej pozwoliły uzyskać komórki, produkujące cenny antybiotyk dla rolnictwa, a dziś środek ten poddawany jest badaniom i czekają go próby fabryczne.

Obecnie w dziedzinie technologii kosmicznej znajdujemy się o krok od produkcji półprzemysłowej...

Po pierwszym wyjściu Aleksieja Leonowa w Kosmos, przez długi czas najważniejsze było opanowanie nawyków sprawnego poruszania się w otwartej przestrzeni. Kosmonauci uczyli się przesuwać wzdłuż konstrukcji, przy tym ćwiczyli sposoby porusza-

W POSZUKIWANIU BLIŹNIAKA ZIEMI

Któż z nas, patrząc w rozgwieżdżone niebo, nie zastanawiał się, czy gdzieś tam daleko, na jakiejś planecie żyją istoty podobne do nas. Proponujemy Ci wspólną wyprawę na poszukiwanie życia w Kosmosie.

Na początek przypomnijmy sobie kilka wiadomości z astronomii:

Planety powstają mniej więcej w tym samym czasie co ich gwiazda centralna — z tego samego wirującego obłoku pyłowego. Obłok taki składa się głównie z wodoru, 20% helu i około 2% pyłu.

Obłok kurczy się grawitacyjnie, równocześnie rośnie jego prędkość obrotowa, w wyniku czego najczęściej rozpada się na części. Każda z nich tworzy gwiazdę. Tak powstają układy podwójne, potrójne itd. Takie systemy mogą być zbyt mało stabilne by umożliwiały powstawanie planet. Jednakże w 10% przypadków, z szybko wirującego fragmentu obłoku tworzy się pojedyncza gwiazda z układem planetarnym. Obłok rozpłaszcza się w dysk, którego centralne zgrubienie tworzy właśnie tę gwiazdę.

Proces kurczenia się ogrzewa dysk i jego centralne części stają się tak gorące, że wyparowują odrobiny lodu a nawet krzem i żelazo. W samym centrum (jeżeli jest tylko wystarczająco gorące) reakcja termojądrowa powoduje powstanie nowej gwiazdy. Sam dysk jednak powoli stygnie. Cząstki żelaza, krzemu i lodu zderzają się i sklejają tworząc ziarna, które przyciągając się już wzajemnie tworzą planety i satelity. Reszta obłoku ucieka w przestrzeń kosmiczną.

Cały ten proces zajmuje od około miliona do dziesiątków milionów lat — niewiele w porównaniu z wynoszącym 4.55 bilionów lat wiekiem układu słonecznego lub 13 bilionów lat naszej galaktyki.

Z najbardziej masywnych obłoków powstają najbardziej masywne gwiazdy, żyjące jednakże najkrócej. Gwiazda o 30-krotnej masie Słońca spala swoje paliwo 150000 razy szybciej wyczerpując je

```
   1 GO TO 1000: REM blizniak zi
emi
 100 REM typy gwiazd
 205 CLS : PRINT
 206 PRINT "              TYP GWIAZD
Y": PRINT
 210 PRINT "TYP        TEMP.  CZAS
 ZYCIA MASA": PRINT : PRINT

 230 PRINT "NIEBIESKI ULTRA  BAR
DZO      30-3"
 240 PRINT "OLBRZYM    FIOLET KRO
TKI": PRINT
 250 PRINT "PRZEJSC-   NIEBIE- KR
OTKI       3-2"
 260 PRINT "IOWY       SKI      ":
 PRINT
 270 PRINT "SLONECZ-   BIALY  DLU
GI      2-0.5"
 280 PRINT "NY
   ": PRINT
 290 PRINT "CZERWONE   CZER-  BAR
DZO  0.5-0.1"
 300 PRINT "KARLY      WONY   DLU
GI         ": PRINT "___________
_____________________"

 315 PRINT : PRINT "JEDNOSTKI MA
SY = MASA SLONCA": PRINT : PRINT

 320 INPUT "MASA GWIAZDY: ";M: I
F M>30 OR M<.1 THEN GO TO 320
 322 LET L=EXP (1.5*LN (M)): PRI
NT : REM jasnosc gwiazdy
 323 LET TA=EXP (-1.1*LN (M))*10
000: REM czas zycia gwiazdy
 324 LET Q=10: LET X=L: GO SUB 9
00: LET L=X: REM zaokraglanie
 325 LET Q=10: LET X=TA: GO SUB
900: LET TA=X: REM zaokraglanie
 328 FOR J=1 TO 63: IF M>S(J) TH
EN GO TO 330: NEXT J
 330 CLS : PRINT "TYP WIDMOWY GW
IAZDY: ";S$(J): PRINT
 332 PRINT "JASNOSC GWIAZDY: ";
L;" RAZY JASNOSC SLONCA": PRINT
 334 PRINT "CZAS ZYCIA GWIAZDY:
";TA;" MLN. LAT": PRINT
 336 IF TA>12000 THEN PRINT "GWI
AZDA MOZE BYC TAK STARA JAK GALA
KTYKA": PRINT
 340 IF TA<100 THEN PRINT T$(5):
 GO TO 350: REM gwiazda moze eks
plodowac
 342 IF TA<700 THEN PRINT T$(1):
 GO TO 350: REM brak form zycia
 344 IF TA<2000 THEN PRINT T$(2)
: GO TO 350: REM brak tlenu w at
mosferze
 346 IF TA<4100 THEN PRINT T$(3)
: GO TO 350: REM moga byc nizsze
 formy zycia
 348 PRINT T$(4): REM moga byc w
yzsze formy zycia
 350 PRINT : PRINT "CZY CHCESZ P
RZEBADAC TEN SYSTEM PLANETARNY?"
 351 PRINT "                (T/N)"

 352 LET Y$=INKEY$: IF Y$="" THE
N GO TO 352
 354 IF Y$="T" OR Y$="t" THEN GO
 SUB 400
 365 FOR W=1 TO 100: NEXT W
 370 PRINT : PRINT "SZUKASZ JESZ
CZE RAZ?"
 380 LET Y$=INKEY$: IF Y$="" THE
N GO TO 380
 385 IF Y$="T" OR Y$="t" THEN GO
 TO 100
 399 STOP
 400 REM planety w systemie
 403 CLS : PRINT : PRINT
 404 LET SA=RND*TA: IF SA>13000
THEN LET SA=13000: REM wiek syst
emu planetarnego
 405 LET Q=10: LET X=SA: GO SUB
900: LET SA=X: REM zaokraglanie
 406 PRINT : PRINT "ODKRYWAMY ZE
 SYSTEM MA ";SA;" MILIONOW LAT"
 407 PRINT : PRINT "(WIEK SYSTEM
U SLONECZNEGO - OKOLO 4550 MILIO
NOW LAT)"
 408 IF SA=13000 THEN PRINT : PR
INT "GWIAZDA DRUGIEJ POPULACJI,
POSIADA MALO CIEZKICH PIERWIASTK
OW"
 409 PRINT
 410 REM orbity
 411 LET NP=6+INT (RND*5): REM l
iczba planet
 412 LET SC=.3*SQR (M)*(.9+RND*.
2)*8/NP: REM skala dysku planeta
rnego
 414 LET A=SC: REM promien pierw
szej orbity
 416 REM zmodyfikowane prawo Bod
e'go
 420 FOR I=1 TO NP
 423 LET A=A*1.7*(.8+RND*.4): RE
M promien I-tej orbity
 425 LET T(I)=SQR (A↑3/M)
 427 LET E(I)=1E-2*RND*9: REM ni
ecentrycznosc orbity
 430 REM masa
 432 LET AC=4*L*L: REM promien s
kraplania sie lodu
 434 LET A2=A↑2: REM promien orb
ity w kwadracie
 435 LET MC=M*(.3+RND*1.4)*A2: R
EM kamieniste jadro
 436 LET B=A/4: LET MC=MC*(.3)↑B
: LET MP=MC: REM pierscien przez
edza sie na brzegach
 437 IF A>AC THEN LET MP=MC*12:
REM lod skrapla sie na pyle
 438 IF MP>3 THEN LET MP=MP↑1.5:
 REM prog grawitacyjny
 439 IF MP>999 THEN LET MP=MP/2:
 GO TO 439
 440 IF MP<1E-2 THEN LET MP=1E-2
 442 REM sklad
 444 LET C$(I)="KAMIENISTY"
 446 IF A>AC THEN LET C$(I)="LOD
OWY"
 448 IF MP>3 THEN LET C$(I)="GAZ
OWY"
 450 REM promien planety
 451 LET R=MC↑.33: REM promien k
amienistego jadra
 452 IF MP<>MC THEN LET R=(MP*5)
↑.33: REM planeta gazowa ma duzy
 promien
 454 REM ciazenie powierzchniowe
 456 LET SG=MP/R/R: REM ciazenie
 powierzchniowe w stosunku do zi
emskiego
 458 LET M(I)=MP: LET A(I)=A: LE
T R(I)=R: LET G(I)=SG: REM zapam
ietanie wartosci
 460 REM satelity
 462 IF MP<2 THEN LET N(I)=INT (
RND+.6): GO TO 490
 464 IF MP>2 THEN LET N(I)=INT (
```

nia się, wzajemnej asekuracji, określali możliwość dokonywania niektórych operacji z przyrządem i przyborami. Po każdym takim wyjściu zbierano cenne informacje uzyskane przez załogę, opracowywano swoiste zasady zachowania się człowieka w otwartym Kosmosie, doskonalono urządzenia i przybory.

Jeden z moich kolegów objaśniając, co znaczy pracować w skafandrze, przytoczył następujące niezwykłe, ale dość trafne porównanie. „Wyobraźcie sobie — powiedział — że znajdujecie się wewnątrz napompowanej gumowej piłki. Po to, by uczynić jakikolwiek ruch, albo wziąć cokolwiek do ręki należy pokonać opór powłoki tej piłki". Jasne jest, że w tym celu potrzebna jest wystarczająca siła i zręczność. Dlatego też, najbardziej niewygodne prace w skafandrze to te, które wymagają niewielkich, ale precyzyjnych ruchów.

Długo można by opowiadać o tym, jak pracuje się dziś w Kosmosie, a jeszcze dłużej — o perspektywach. Zdaniem specjalistów, takie globalne problemy jak energetyczny, ochrona środowiska naturalnego, prognozowanie i sterowanie pogodą zostaną pomyślnie rozstrzygnięte dzięki wykorzystaniu kosmonautyki. Z każdym rokiem coraz wyraźniej widać, ile racji miał genialny Ciołkowski w swych prognozach na temat wykorzystania lotów kosmicznych w interesie ludzkości. Przecież obecne kompleksy orbitalne typu „Salut-Sojuz", to pierwowzory wspaniałych „kosmicznych osad", o których marzył. Dziś już nie odległym marzeniem, a bliską rzeczywistością są plany stworzenia kosmicznych wydziałów produkcyjnych i fabryk. Ich prototypy — eksperymentalne linie technologiczne — już wiele lat pomyślnie funkcjonują na orbitach. Kosmonautyka weszła dziś na stałe w system zapewnienia Ziemi przekazu radio-telewizyjnego. Wspólnym wysiłkiem specjalistów 4 krajów — ZSRR, USA, Kanady, Francji — stworzony został kosmiczny system poszukiwania zaginionych statków i samolotów („Kospas-Sarsat"). Uratowano dzięki niemu setki istnień ludzkich. Najbardziej interesujące i trudne międzynarodowe projekty zbadania wszechświata — takie np. jak „Wega" czy „Fobos" — pomagają specjalistom rozszerzać i pogłębiać naszą wiedzę o otaczającym świecie, o ewolucji gwiazd i galaktyk, o procesie życia na Ziemi, o historii i przyszłości systemu słonecznego.

W istocie horyzonty działalności ludzkiej w Kosmosie są nieograniczone. Im szersza i głębsza jest nasza wiedza o nim, tym więcej nowych problemów stawia on przed ludźmi i od nich rozwiązania pod wieloma względami zależy dalszy rozwój cywilizacji. Ciołkowski w swych naukowych rozmyślaniach o Kosmosie marzył, że przyniesie on ludzkości góry chleba i bezmiar potęgi. I rzeczywiście: osiągnięcia kosmonautyki służą dziś gospodarce i kulturze, pomagają rozwiązywać ważne problemy rolnictwa i medycyny, geologii i geografii, transportu i łączności.

Opr. W. S.



w ciągu około jednego miliona lat w odróżnieniu od Słońca, które będzie istnieć 10 bilionów lat. Gwiazda ciężka może także skończyć jako supernowa, gwiazda neutronowa itp.

Tworzenie planet zależy od temperatury części gazowego obłoku, a to z kolei zależy od odległości planety od gwiazdy centralnej. Wewnętrzna część dysku planetarnego jest zbyt gorąca by utrzymał się lód, tam zatem powstają planety kamieniste — zbudowane z gęstego żelaznego jądra i krzemowego płaszcza powierzchniowego. Dalsze planety są większe lecz mniej gęste, złożone są głównie z gazu i lodu. (Granica między planetami skalnymi i lodowo-gazowymi wyznaczana jest przez promień skraplania lodów. W naszym układzie wynosi 4 J.A. (jednostki astronomiczne — odległość Ziemi od Słońca).

W listopadowym numerze „Creative Computing" z zeszłego roku znalazł się artykuł H. Carlsona p.t. „W poszukiwaniu bliźniaka Ziemi". Autor prezentuje program umożliwiający prosty sposób przebadania wielu układów planetarnych.

Wybieramy masę gwiazdy, której układ chcemy zbadać. Program tworzy — w części losowo-typowy system planetarny dla tej gwiazdy, korzystając z obecnego rozumienia praw tworzenia się układów planetarnych (są one lepiej poznane dla gwiazd o masie Słońca — M=1). Systemy tworzone dla gwiazd o innych masach są właściwie losowe.

Promienie orbit wyznaczone są ze zmodyfikowanego prawa Titius-Bode'go — promienie rosną średnio w postępie geometrycznym na zewnątrz od gwiazdy. Założono, że masa dysku planetarnego jest proporcjonalna do masy gwiazdy centralnej. Uwzględniając odwrotnie kwadratową zależność natężenia światła od odległości od gwiazdy do wyznaczenia promienia skraplania lodu i stosując prawa Keplera można już zbudować układ planetarny.

Przekonasz się szybko, że jest bardzo mało planet pasujących do Ziemi pod względem temperatury, siły przyciągania i składu chemicznego. Jeżeli tylko promień orbity jest trochę większy lub mniejszy od ziemskiego to temperatura będzie za niska lub za wysoka. Jeżeli siła grawitacji będzie trochę mniejsza to ucieknie planecie atmosfera, a jeżeli większa to jej skład będzie się znacznie różnił od 80% azotu i 20% tlenu. Te warunki będziesz mógł sprawdzić, jeśli samodzielnie rozszerzysz program. Spróbuj też zakończyć rozpoczętą konstrukcję satelitów. Ich pochodzenie może być dwojakiego rodzaju: mogą powstać z dysków pyłowych krążących wokół dużych gazowych planet (tak jak wokół gwiazdy), lub mogą być wyłapane z przestrzeni kosmicznej.

Jeżeli lubisz chemię, zajmij się składem planety, przejdź do geologii, później do pogody i klimatu. W końcu zastanów się jaki to wszystko ma wpływ na biologię planety. Z pewnością przyda Ci się książka Astronomia Popularna, pod redakcją Stefana Piotrowskiego.

Przyślij do nas swój program, opisz jaką przyjąłeś metodykę obliczeń i z jakiej literatury korzystałeś. Dla autorów najciekawszych rozwiązań mamy nagrody.

Prezentowany przez nas program przeznaczony jest na ZX Spectrum, jeśli jednak masz już jakieś doświadczenie, bez trudności zapiszesz go na każdym innym komputerze.

Bajtek

```
5.6*RND+RND*2.2*LN (MP))
 490 NEXT I
 500 REM geofizyczne warunki pla
nety
 510 REM geometria orbit
 520 FOR I=1 TO NP
 524 LET I(I)=.1*RND*3: REM nach
ylenie osi planety
 526 LET D(I)=1: REM dlugosc dni
a (do ustalenia)
 528 LET O(I)=290*SQR (SQR (L)/A
(I)): REM temperatura powierzchn
iowa planety
 530 IF O(I)>173 AND A<=AC AND M
P<=3 THEN LET M(I)=M(I)/12: LET
C$(I)="KAMIENISTY"
 532 IF O(I)>2000 THEN LET M(I)=
0: LET C$(I)="WYPAROWALA"
 538>NEXT I: GO SUB 700: REM dru
kuj tabelke
 550 RETURN
 700 REM drukuj tabelke planetar
na
 712 CLS : PRINT : PRINT : PRINT
"            PLANETY": PRINT
 730 FOR I=1 TO NP
 732 LET Q=10: LET X=A(I): GO SU
B 900: LET A(I)=X: REM zaokragla
nie
 734 LET Q=10: LET X=G(I): GO SU
B 900: LET G(I)=X: REM zaokragla
nie
 736 LET Q=10: LET X=M(I): GO SU
B 900: LET M(I)=X: REM zaokragla
nie
 738 LET Q=10: LET X=O(I): GO SU
B 900: LET O(I)=X-270: REM zaokr
aglanie
 740 LET Q=10: LET X=T(I): GO SU
B 900: LET T(I)=X: REM zaokragla
nie
 742 PRINT "PLANETA
"; I: IF O(I)>2000 THEN PRINT C$
(I): GO TO 770
 743 IF C$(I)="WYPAROWALA" THEN
PRINT C$(I): GO TO 770
 744 PRINT "PROMIEN ORBITY
";A(I);" J.A.": REM promien orb
ity w jednostkach astronomicznyc
h
 748 PRINT "OKRES
";T(I);" LAT"
 752 PRINT "MASA
"; M(I);" ZIEM."
 756 PRINT "PRZYCIAGANIE POW.
";G(I);" ZIEM."
 760 PRINT "TEMPERATURA POW.
";O(I);" C"
 766 PRINT "LICZBA KSIEZYCOW
";N(I)
 768 PRINT "SKLAD POWIERZCHNI
";C$(I)
 769 PRINT "NACISNIJ DOWOLNY KLA
WISZ": PAUSE 0: CLS
 770 NEXT I
 799 RETURN
 900 REM procedura zaokraglajaca
 906 IF X=0 THEN RETURN
 910 LET N=0: IF X<2*Q THEN GO T
O 930
 914 IF X>=20*Q THEN LET X=X/10:
 LET N=N+1: GO TO 914
 918 LET X=INT (X+.5): IF N=0 TH
EN RETURN
 920 FOR K=1 TO N: LET X=X*10: N
EXT K: RETURN
 930 IF X<2*Q THEN LET X=X*10: L
ET N=N+1: GO TO 932
 934 LET x=INT (x+.5): IF N=0 TH
EN RETURN
 936 FOR K=1 TO N: LET X=X/10: N
EXT K: RETURN
1000 REM opis
1005 CLS : PRINT : PRINT : PRINT

1020 PRINT "           BLIZNIAK ZIE
MI": PRINT : PRINT
1030 PRINT "  PRZEMIERZAMY KOSMO
S W POSZUKI-WANIU BLIZNIAKA ZIEM
I - PLANETY";
1031 PRINT " KTORA MOZEMY SKOLON
IZOWAC."
1032 PRINT "  WYBIERAMY GWIAZDE
CENTRALNA O POZADANEJ MASIE"
1099 PRINT
1100 PRINT AT 17.4;"NACISNIJ KLA
WISZ SPACJI"
1110 LET Y$=INKEY$: IF Y$<>" " T
HEN GO TO 1110
1999 PRINT : PRINT "
CZEKAJ"
2000 REM inicjalizacja
2120>DIM O(11): DIM S(65): DIM S
$(65,2): DIM T$(5,70): DIM T(11)
: DIM E(11): DIM M(11): DIM A(11
): DIM R(11): DIM G(11): DIM N(1
1): DIM I(11): DIM D(11): DIM C$
(11,70)
2125 RESTORE 2160
2160 DATA 30,26,23,19.9,17.3,15.
1,13.2,11.5,10,8.4,7.1,6,5,4.2,3
.6,3,2.9,2.7,2.6,2.5,2.3
2161 DATA 2.2,2.1,2,1.91,1.82,1.
73,1.65,1.57,1.49,1.42,1.35,1.28
,1.22,1.16,1.11,1.05,1,.95,.9,.8
5,.81,.77,.73,.69,.66,.62,.59,.5
6,.53,.51,.48,.46,.43,.41
2162 DATA .39,.37,.35,.33,.25,.1
63,.135,.1
2164 DATA "O5","O6","O7","O8","O
9","B0","B1","B2","B3","B4","B5"
,"B6","B7","B8","B9","A0","A1","
A2","A3","A4","A5","A6","A7","A8
","A9"
2165 DATA "F0","F1","F2","F3","F
4","F5","F6","F7","F8","F9","G0"
,"G1","G2","G3","G4","G5","G6","
G7","G8","G9"
2166 DATA "K0","K1","K2","K3","K
4","K5","K6","K7","K8","K9","M0"
,"M1","M2","M3","M4","M5","M6","
M7"
2168 FOR J=1 TO 63: READ S(J): N
EXT J
2169 FOR J=1 TO 63: READ S$(J):
NEXT J
2180 LET T$(5)="GWIAZDA MOZE EKS
PLODOWAC W NAJBLIZSZEJ PRZYSZLOS
CI !"
2181 LET T$(1)="SYSTEM JEST ZA M
LODY BY MOGLO SIE ROZWINAC ZYCIE
"
2182 LET T$(2)="MOZE ISTNIEC ZYC
IE KOMORKOWE, ALE NIE MA TLENU N
A ZADNEJ PLANECIE"
2183 LET T$(3)="MOZE BYC TLEN I
NIZSZE FORMY ZYCIA"
2184 LET T$(4)="MOGA BYC WYZSZE
FORMY ZYCIA NA NIEKTORYCH PLANET
ACH"
2999 GO TO 100: REM skok do czes
ci glownej programu
```

HORYZONTY KOSMOSU

NIE TYLKO KOMPUTERY

Komputer jest ważny ale... nie jest wszystkim. Dlatego uznaliśmy, że źle byłoby gdyby w pierwszym w Polsce piśmie określającym się „przepustką w XXI wiek" zabrakło miejsca na wielkie problemy PRZYSZŁOŚCI. A przyszłość to nie tylko komputery, ale również opanowanie kosmosu, zagospodarowanie oceanicznych głębin, automatyzacja i robotyzacja, energetyka termojądrowa, biotechnologie, klonowanie, edukacja. Postęp w tych wszystkich dziedzinach możliwy jest tylko dzięki rozwojowi techniki komputerowej — to oczywiste. Ale, z drugiej strony, rozwój wszystkich tych dziedzin, stawiając nowe wymagania, wpływa na rozwój informatyki. Warto zdawać sobie sprawę z tego sprzężenia zwrotnego.

Dzisiaj prezentujemy refleksje młodej radzieckiej kosmonautki **SWIETŁANY SAWICKIEJ** nt. możliwości, jakie otwiera człowiekowi przestrzeń kosmiczna:

Pierwszy sputnik, pierwszy lot po orbicie wokółziemskiej Jurija Gagarina i pierwsze wyjście w otwarty Kosmos Aleksieja Leonowa — to trzy wydarzenia, które i dziś, po dziesiątkach lat, są najważniejszymi w historii kosmonautyki.

W ostatnich latach na świecie rozwiązano wiele skomplikowanych zadań naukowych i technicznych. Za każdym razem rozwiązania te demonstrowały potęgę wspólnej pracy dużej liczby specjalistów, złączonych wspólnym celem, ogromne możliwości umysłów i charakterów wielu ludzi, którzy wspólnym wysiłkiem realizowali coraz to nowe ambitne zamysły, oznaczające konsekwentne wspinanie się człowieka po dro-

Przyszłość kosmonautyki rysuje się o wiele ciekawiej niż najbardziej nawet śmiałe pomysły z zakresu „science fiction".